

VŠB – Technická univerzita Ostrava

Fakulta strojní

Katedra automatizační techniky a řízení

# **Webová informační podpora benchmarkingu vysokých škol**

*Web-oriented Information Support for the University Benchmarking Projekt*

Student: Jana Ježíková

Vedoucí diplomové práce: prof. Ing. Radim Farana, Csc.

Konzultantka: Ing. Jitka Smutná

Ostrava 2010

## Zadání diplomové práce

---

Posluchač:	Bc. Jana Ježíková (JANICKAJEZECEK@seznam.cz)
Studijní obor :	3902T004-00 Automatické řízení a inženýrská informatika
Název tématu :	Webová informační podpora benchmarkingu vysokých škol <i>Web-oriented Information Support for the University Benchmarking Project</i>
Akademický rok :	2008/2009
Vedoucí diplomové práce :	prof. Ing. Radim Farana, CSc.
Konzultantka:	Ing. Jitka Smutná

### ZADÁNÍ :

1. Popište stručně možnosti současných webových nástrojů a databázových systémů s důrazem na jejich využitím pro vytvářený informační systém.
2. Analyzujte potřeby benchmarkingového projektu strojních fakult ČR a SR. Vytvořte soubor typických používaných ukazatelů a metodiku jejich určení.
3. Realizujte datovou strukturu informačního systému pro podporu realizace benchmarkingového projektu a ověřte její funkcionalitu na vhodných testovacích datech.
4. Navrhněte a realizujte strukturu aplikační vrstvy vytvářeného informačního systému, ověřte správnost zvolené koncepce na vhodném souboru testovacích, případně ji podle dosažených výsledků upravte.
5. Vytvořte prezentační vrstvu systému pro grafické zobrazení výsledků benchmarkingu strojních fakult ČR a SR.

## DOPORUČENÁ LITERATURA:

Česká společnost pro jakost. Benchmarkingový server. [on-line]. cit. 2008-03-05. Dostupný z webu  
<URL: <http://www.benchmarking.cz/>

FARANA, R. *Podklady pro benchmarking strojních fakult vysokých škol*. [MS-Excel]. Datum poslední změny 2008-02-04 .

FARANA, R. *Databáze – speciální postupy*. 1. vyd. Praha : ČSVTS, SmSVTSaP, KAKI, 2006, 170 s. ISBN 80-02-01876-1.

HELD, B. 2006. *Access VBA : velká kniha řešení*. 1. vyd. Brno : Computer Press, 2006, 639 s. ISBN 80-251-1112-1.

SMITH, R. & SUSSMAN D. *Beginning Access 2000 VBA*. 1<sup>st</sup> ed. Chicago, IL, USA : Wrox Press Ltd. 2000, 866 pp. ISBN 1-861001-76-2.

ŠARMANOVÁ, J. *Datové sklady a dolování znalostí z nich*. Ostrava : FEI VŠB-TU Ostrava, 2003.

ŠLOSARČÍK, M. *Informační podpora benchmarkingu vysokých škol. Semestrální projekt 2*. Ostrava: VŠB-TU Ostrava [PDF], datum poslední změny: 2009-02-16.

ŽÍDEK, Adam. *Databázová podpora měření*. VŠB-TUO, katedra ATR-352 VŠB-TUO, 2005. 36 s. Bakalářská práce, vedoucí: Farana, R.

V Ostravě 18. 2. 2010

prof. Ing. Radim Farana, CSc.

**Prohlášení studenta**

Prohlašuji, že jsem celou diplomovou práci včetně příloh vypracovala samostatně pod vedením vedoucího diplomové práce a uvedla jsem všechny podklady a literaturu.

V Ostravě: 21. 5. 2010

.....

Plné jméno studenta

**Prohlašuji, že**

- byla jsem seznámena s tím, že na moji diplomovou práci se plně vztahuje zákon č.121/2000 Sb. – autorský zákon, zejména §35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a §60 – školní dílo.
- беру на ве́домі́, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen VŠB-TUO) má právo nevýdělečně ke své vnitřní potřebě diplomovou práci užít (§35 odst. 3).
- souhlasím s tím, že jeden výtisk diplomové práce bude uložen v Ústřední knihovně VŠB-TUO k prezentačnímu nahlédnutí a jeden výtisk bude uložen u vedoucího diplomové práce. Souhlasím s tím, že údaje o diplomové práci budou zveřejněny v informačním systému VŠB-TUO.
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu §12 odst. 4 autorského zákona.
- bylo sjednáno, že užít své dílo – diplomovou práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu náklad, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).
- беру на ве́домі́, že odevzdáním své práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby.

V Ostravě: 21. 5. 2010

Dobrá č. 71

739 51 Dobrá

.....

## ANOTACE DIPLOMOVÉ PRÁCE

Ježíková, J. *Webová informační podpora benchmarkingu vysokých škol*. Ostrava: kat. ATŘ – 352 VŠB-TUO, 2009. 109 s. Diplomová práce, vedoucí: Farana, R.

Diplomová práce se zabývá tvorbou webového informačního systému k podpoře benchmarkingu vysokých škol. V úvodní části je seznámení s pojmy benchmarkingu, dále je uveden rozbor ukazatelů pro benchmarking včetně popisů jejich výpočtů. Pro realizaci tohoto systému bylo nutno vytvořit databázi, ta je tvořena v SQL Serveru 2008, její struktura a výhody tohoto programu jsou zde také popsány. Hlavní část se ale věnuje návrhu a tvorbě informačního systému v programu ASP.NET. Také je zde lehké seznámení s tímto programem včetně popisu jeho prostředí. V závěrečné části je rozebráno zabezpečení systému a manuál, který podrobně popisuje celý postup práce s informačním systémem. Součástí závěru je také kritické zhodnocení dosažených výsledků a návrh případného dalšího vylepšení webového systému.

## ANNOTATION OF THE DIPLOM THESIS

Ježíková, J. *Web-oriented Information Support for the University Benchmarking Projekt*. Ostrava: Department. of Control Systems and Instrumentation, Faculty of mechanical Engineering VŠB – Technical University of Ostrava, 2009, 109 p. Diplom thesis, head: Farana, R.

This thesis deals with creating a web information system to support benchmarking universities. The first part is to introduce the concepts of benchmarking, then there is analysis of indicators for benchmarking, including descriptions of their calculations. To realize this system it was necessary to create a database that is created in SQL Server 2008, its structure and benefits of this program are also described here. But the main part deals with design and production of information system in the ASP.NET. There is also a slight acquaintance with this program, including a description of its environment. The final section analyzes the security of the system and a manual which describes in detail the whole process of working with information system. The conclusion contains also a critical evaluation of results achieved and a concept of any further improvements of the web system.

# Obsah

Seznam použitých zkratk a značení .....	11
Úvod .....	14
1 Benchmarking .....	15
1.1 Podstata a smysl benchmarkingu .....	15
1.2 Benchmarkingový cyklus.....	16
1.3 Druhy benchmarkingu.....	16
1.4 Kodex benchmarkingu .....	17
1.5 Oblasti pro benchmarkingový sběr dat .....	17
1.6 Ukazatelé a jejich definice .....	19
2 Informační systém .....	21
2.1 Pojem informační systém.....	21
2.2 Výhody a nevýhody customizovaného IS.....	21
2.3 Proces návrhu systému.....	22
2.4 Databáze a databázové systémy .....	24
2.4.1 Hierarchický model .....	24
2.4.2 Síťový model .....	25
2.4.3 Relační model .....	25
2.4.4 Dotazovací jazyk .....	25
2.5 Současné databázové systémy .....	26
2.5.1 MySQL .....	26
2.5.2 Oracle.....	26
2.5.3 Microsoft Access .....	27
2.5.4 FoxPro .....	27
2.5.5 Microsoft SQL Server.....	27
3 Návrh a ověření datové struktury .....	28
4 Návrh struktury aplikační vrstvy IS .....	34
5 Prostředí pro tvorbu aplikační vrstvy .....	37
6 Připojení databáze ve Visual Studiu 2008.....	40

7	Mapa stránek, tvorba šablony.....	43
7.2	Tvorba šablony.....	44
8	Registrace, přihlášení, změna hesla uživatele .....	47
9	Seznam škol, úprava a odstranění .....	51
10	Vkládání dat .....	53
11	Nástroje pro tvorbu grafů.....	55
11.1	Fusion charts .....	55
11.2	Google Chart API.....	55
11.3	Xceed Chart for ASP.net.....	56
11.4	Visifire .....	56
11.5	Tee chart.....	57
11.6	Microsoft Chart Controls .....	57
11.6.1	Vykreslování grafů v MS Chart Control.....	58
11.6.2	Tabulkové a grafické srovnání.....	59
12	Závěr .....	61
	Použitá literatura.....	63

## Seznam obrázků

Obr. 1	Proces benchmarkingu .....	15
Obr. 2	Obrazovka programu MS SQL Server 2005 včetně seznamu tabulek .....	28
Obr. 3	Tabulka škola v návrhovém prostředí .....	28
Obr. 4	Tabulka ukazatelů v návrhovém zobrazení .....	29
Obr. 5	Tabulka vstup v návrhovém zobrazení.....	29
Obr. 6	Tabulka výstup v návrhovém zobrazení.....	29
Obr. 7	Tabulka výpočet v návrhovém zobrazení.....	30
Obr. 9	Pokračování tabulky škola naplněné daty .....	30
Obr. 10	Seznam škol s kontaktními osobami .....	31
Obr. 11	Seznam škol.....	31
Obr. 12	Seznam škol v jednotlivých letech .....	31
Obr. 13	Seznam škol v jednotlivých letech v sql jazyce .....	32
Obr. 14	Hodnoty výstupů ukazatelů pro jednotlivé školy - sql zápis.....	32
Obr. 15	Hodnoty výstupů ukazatelů pro jednotlivé školy .....	32
Obr. 16	Datová struktura databáze .....	33



Obr. 17 Úvodní obrazovka benchmarkingu .....	34
Obr. 18 Registrační formulář.....	34
Obr. 19 Potvrzení registrace .....	35
Obr. 20 Formulář srovnávací analýzy .....	35
Obr. 21 Formulář vyhodnocení .....	36
Obr. 22 Formulář zadávání údajů.....	36
Obr. 23 Přepisovací dialogové okno.....	36
Obr. 24 Vytvoření webové aplikace .....	38
Obr. 25 Prostředí ASP.NET .....	39
Obr. 27 Připojení databáze v design režimu .....	40
Obr. 28 Druh databáze.....	41
Obr. 29 Výběr serveru a příslušné databáze .....	41
Obr. 30 Výběr tabulky a jejich sloupců.....	42
Obr. 32 Přidání šablony.....	44
Obr. 33 Jednotná definice názvů stránek.....	44
Obr. 34 Části šablony .....	45
Obr. 35 Vzhled šablony .....	46
Obr. 36 Přidání nové stránky se vzhledem šablony.....	47
Obr. 37 Připojení dat z databáze.....	47
Obr. 38 Registrace .....	48
Obr. 39 Procedura SendEmail .....	49
Obr. 40 Přihlášení.....	50
Obr. 41 Změna hesla.....	50
Obr. 42 Náhled stránky Skola.aspx .....	51
Obr. 43 Ukázka vyhledávání .....	51
Obr. 44 Potvrzení smazání uživatele .....	52
Obr. 45 Náhled editace uživatele.....	52
Obr. 46 Procedura sp_Vstup.....	53
Obr. 47 Definice výpočtu ukazatele s identifikačním číslem 4 .....	54
Obr. 48 Stránka Vkladani.aspx.....	54
Obr. 49 Graf v programu Visifire.....	56
Obr. 50 Ukázka grafu v MSChart .....	57
Obr. 51 Komponenta Chart v nabídce nástrojů .....	58
Obr. 52 Připojení databáze k grafu.....	58
Obr. 53 Menu vlastnosti komponenty chart .....	58
Obr. 54 Kritéria pro srovnání .....	59
Obr. 55 Tabulka vybraných dat .....	59

Obr. 56 Výsledné zobrazení grafu.....	60
---------------------------------------	----

### **Seznam tabulek**

Tab. 1 Seznam pokynů pro tvorbu benchmarkingu.....	16
Tab. 2 Základní kroky analýzy a návrhu ISS .....	22
Tab. 3 Projektový záměr.....	23
Tab. 4 Příklady regulárních výrazů .....	49

## Seznam použitých zkratek a značení

<b>.NET</b>	(„dotnet“) soubor technologií tvořící celou platformu. Základní komponent je MS.NET Framework. Pro vývoj .NET aplikací vydal MS Visual Studio .NET.
<b>ANSI</b>	americký standardizační institut
<b>Apache Server</b>	je softwarový webový server s otevřeným kódem pro Linux, BSD, Microsoft Windows a další platformy. V současné době dodává prohlížečům na celém světě většinu internetových stránek.
<b>ASP ( Active Server Pages)</b>	Skriptovací platforma určená pro dynamické zpracování webových stránek na straně serveru.
<b>ASP.NET</b>	Nástupce ASP, širší a komplexnější technologie, která se od ASP v mnoha ohledech liší.
<b>C</b>	programovací jazyk, který byl vyvinut pro potřeby operačního systému Unix. V současné době je to jeden z nejpopulárnějších jazyků, zřejmě nejčastější pro psaní systémového softwaru, ale velmi rozšířený i pro aplikace.
<b>C++</b>	je objektově orientovaný programovací jazyk, rozšířením jazyka C.
<b>CSS ( Cascading Style Sheets )</b>	Kaskádové styly. Je to jazyk pro popis způsobu zobrazení stránek napsaných v jazycích HTML, XHTML nebo XML.
<b>Framework</b>	Softwarová struktura, která slouží jako podpora při programování a vývoji projektů.
<b>HTML ( HyperText Markup Language )</b>	Značkovací jazyk, je jedním z jazyků pro vytváření stránek v systému www (World Wide Web), který umožňuje publikaci na internetu.
<b>IBM</b>	(International Business Machines Corporation) je přední světová společnost v oboru informačních technologií.
<b>IS</b>	informační systém

<b>Java</b>	objektově orientovaný programovací jazyk, je jedním z nejpoužívanějších programovacích jazyků na světě se skvělou přenositelností
<b>JavaScript</b>	Multiplatformní, objektově orientovaný skriptovací jazyk pro www stránky, často vkládaný přímo do HTML kódu stránky. Jsou jím ovládány různé interaktivní prvky.
<b>Linux</b>	operační unixový systém
<b>Macintosh</b>	známý jako Mac, osobní počítače od Apple Computer.
<b>Microsoft Office</b>	kancelářský balík od firmy Microsoft
<b>MS-DOS</b>	operační systém firmy Microsoft, první operační systém určený pro jednoduchou obsluhu
<b>Perl</b>	interpretovaný programovací, s rozvojem internetu se Perl stal velmi populárním nástrojem pro tvorbu CGI skriptů.
<b>PHP</b>	(Personal Home Page) je skriptovací programovací jazyk pro programování dynamických internet. stránek.
<b>Python</b>	interpretovaný objektově orientovaný programovací jazyk, který je vyvíjen jako open source projekt, který zdarma nabízí instalační balíky pro většinu běžných platform.
<b>QBE</b>	(Query By Example)
<b>SCP</b>	stanovení cílů projektu
<b>SEQUEL</b>	(Structured English Query Language)
<b>SQL</b>	(Structured Query Language) standardizovaný dotazovací jazyk používaný pro práci s daty v relačních databázích.
<b>SQL-86</b>	Norma upravující SQL jazyk, zavedena v roce 1986
<b>SQL-92</b>	Norma upravující SQL jazyky, přijata roku 1992
<b>ŠŘBD</b>	databázový systém, systém řízení báze dat, Database Management System (DBMS)

<b>Tcl</b>	programovací jazyk založený na Lispu
<b>T-SQL</b>	(Transact-SQL) je rozšíření jazyka SQL od společností Microsoft a Sybase. Microsoft tento jazyk používá v produktu Microsoft SQL Server. Sybase Software v Adaptive Server Enterprise.
<b>Unix</b>	ochranná známka operačního systému vytvořeného v roce 1969 americkou firmou AT&T. Unix v současné době vlastní konsorcium The Open Group.
<b>Visual Basic</b>	dialektem programovacího jazyka BASIC od společnosti Microsoft.
<b>Webhosting</b>	Pronájem prostoru pro webové stránky na cizím serveru.
<b>XML</b>	(eXtensible Markup Language, česky rozšiřitelný značkovací jazyk) je obecný značkovací jazyk, který umožňuje snadné vytváření konkrétních značkovacích jazyků pro různé účely a široké spektrum různých typů dat.

## Úvod

Cílem diplomové práce je vytvořit webový informační systém pro benchmarking vysokých škol. Aplikace bude umístěna na webu a bude sloužit k přihlašování do benchmarking systému, k zadávání jednotlivých údajů potřebných pro benchmarking, k jejich vyhodnocování a zobrazování jednotlivých ukazatelů pomocí grafů a tabulek. K tvorbě bude nutno vytvořit analýzu dat pro benchmarking, zvolit některý z nabídky databázových systémů a vytvořit webové rozhraní tohoto systému.

Nejprve se seznámím s pojmem a funkcí benchmarkingu, na jehož základě pak navrhu ukazatele, jejich propojení a výpočet. Abych nezanedbala analýzu vytvořím také projektový záměr, ve kterém ještě jednou stanovím cíle projektu. Na závěr analýzy vytvořím vhodnou datovou strukturu pro tvorbu informačního systému, tj. data benchmarkingu uspořádám do tabulek a vytvořím entitně-relační model.

Dalším důležitým úkolem bude zvolit si z nabídky programů vhodný databázový systém. Před jeho zvolením nejprve vysvětlím pojem databázových systémů, popíšu výhody a nevýhody nejznámějších a nejpoužívanějších z nich. Dále popíši vybraný systém, jeho prostředí, práci a tvorbu databáze v něm. Pomocí již dříve vytvořené datové struktury, v něm poté vytvořím databázi, kterou naplním zkušebními daty a budu ověřovat její funkčnost. Pokud nebude databáze splňovat předpoklady, které jsme si vytyčili v projektovém záměru, přepracuji tabulky a vytvořím novou databázi, jejíž funkčnost opět ověřím.

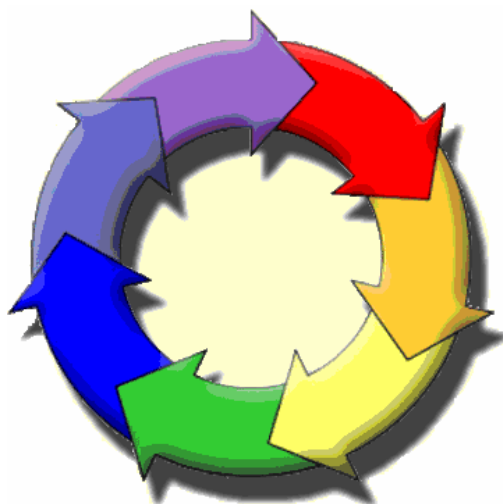
Poté, co dokončím a ověřím strukturu databáze, se budu zabývat návrhem aplikační vrstvy a její realizací, tj. připojením databáze a vytvořením několika jednoduchých formulářů, na kterých rovněž ověřím funkčnost. Dále vytvořím několik informační stránek týkajících se benchmarkingu, stránku pro registraci, přihlášení, změnu hesla, vkládání dat a srovnání dat. Poté zvolím některou z volně přístupných šablon webu a budu se zabývat designem celé aplikace. Po dokončení bude systém znovu otestován v různých prohlížečích a vytvořený projekt bude předán k používání určeným pracovníkům katedry Automatizační techniky a řízení. Závěrem vytvořím manuál pro práci s informačním systémem a navrhu postup dalšího vylepšení.

# 1 Benchmarking

Pro pochopení analýzy dat a tvorby datové struktury pro tvořený informační systém, musíme nejprve vysvětlit, co pod pojmem benchmarking chápeme. Benchmarking (anglicky bench - laťka a marking – označení), poprvé s ním přišla firma Xerox Corporation na poč. 80.let 20. století, jako s nástrojem strategického managementu.

## 1.1 Podstata a smysl benchmarkingu

Někdy se pojem benchmarking vysvětluje jako srovnávací analýza dat, údajů, nákladů a ekonomických výsledků, jiní jej označují za výměnu zkušeností, např. na strategické úrovni řízení organizací. V této semestrální práci budeme chápat benchmarking jako strukturované porovnávání s cílem definovat a zlepšovat dobré praktické postupy na principu učení se od druhých. Tento proces se pro zdokonalování organizace stále opakuje.



**Obr. 1 Proces benchmarkingu [Vzdělávací centrum pro veřejnou správu, 2005]**

Smyslem benchmarkingu je zjistit, (pokud jde například o náklady, počet zaměstnanců, počet studentů, čas, ...) jak si škola stojí ve srovnání s ostatními, jak efektivně pracuje. Vedle zjišťování nedostatků, využijeme proces benchmarkingu k tomu, abychom zjistili, jak některé věci dělají jiné školy, a abychom na základě těchto informací byli schopni vytvořit si svůj názor na to, jak naše služby zdokonalit.

Proč je benchmarking důležitý? Co neumíme měřit, nedokážeme ani efektivně řídit. Benchmarking dává podklady pro kvalifikované rozhodování o změnách, umožňuje se neustále zlepšovat a je-li prováděn dlouhodobě, představuje analýzu časových dat a trendů.

## 1.2 Benchmarkingový cyklus

Tak jako se například každodenní cyklus člověka skládá z neustále se opakujících se činností jako je vstávání, jezení, pracování, hygieny, spaní, tak i proces benchmarkingu má dán svůj cyklus, jehož kroky bychom mohli chápat jako pokyny pro tvorbu benchmarkingu, viz. Tab. 1. Tyto kroky mezi sebou nelze příliš zaměňovat.

Tab. 1 Seznam pokynů pro tvorbu benchmarkingu

Číslo kroku	Popis
1	vybrat oblasti, služby pro benchmarkingové porovnávání
2	definice dat a ukazatelů
3	sběr a zpracování dat
4	analýza dat, identifikace nejlepších výkonů
5	definice nejlepších praktických postupů
6	návrh strategie pro optimalizaci procesů a služeb
7	opakovaná analýza dosažených výsledků

Ze všech uvedených kroků považuji výběr pro porovnávání a návrh ukazatelů jako nejdůležitější části tvorby, jelikož se špatně navrženou strukturou ukazatelů se dále může celý tvořený informační systém sesypat jako “domeček z karet”. Proto bychom tento proces neměli brát na lehkou váhu, naopak bychom na něj měli dávat velký důraz. [Martin Růžička, 2002]

## 1.3 Druhy benchmarkingu

Kvalitu vysokých škol můžeme posuzovat ze spousty hledisek, stejně tak existuje několik druhů benchmarkingu :

- **benchmarking výkonů** (srovnávání klíčových finančních a dalších ukazatelů)
- **benchmarking procesů** (srovnání procesů)
- **benchmarking nejlepších postupů** (srovnání s nejlepšími organizacemi v místním, celostátním, nebo mezinárodním měřítku)

Podle toho, odkud organizace bere srovnávací data, rozlišujeme ještě 2 typy benchmarkingu a to interní a externí:

- **interní** (samotné srovnání jsou hledány uvnitř organizace u útvarů, jež zvládají podobné úkoly lépe, vyskytuje se u velkých organizací)
- **externí** (data jsou čerpány od jiných subjektů, problémem se stává hledání legální cesty převzetí informací - ochrana průmyslových práv.
- **odvětvový** (hledáme u konkurence - především, co se týká výroby)
- **mimoodvětvový** (hledáme mimo konkurenci - týká se např. způsobu archivace dat, komunikace v organizaci apod.).



V mé práci se budu zabývat benchmarkingem externím, výkonů, procesů i benchmarkingem nejlepších postupů. [Czech invest, 2006]

## 1.4 Kodex benchmarkingu

Benchmarking je založen na principu každý vyhrává a podléhá světově uznávanému kodexu vedení benchmarkingu. Hlavní části kodexu zahrnují následující zásady:

- zachovávejte legálnost
- buďte ochotní dát to, co jste dostali
- respektujte důvěrnost
- informace udržujte jako interní
- využívejte kontaktních osob pro benchmarking
- neodkazujte se bez předchozího svolení
- buďte připraveni jít od počátku
- snažte se porozumět očekáváním
- jednejte v souladu s očekáváním
- buďte čestní
- dodržujte závazky

Tento výčet pravidel není ani zdaleka úplný, má sloužit pouze k představě, co daný kodex vedení představuje. Kodex je soubor pravidel, které byly vytvořeny s cílem usměrňovat úsilí o benchmarking a prosazovat profesionalitu a efektivnost benchmarkingu. Dodržování tohoto kodexu přispívá k efektivnímu, účinnému a etickému benchmarkingu.

## 1.5 Oblasti pro benchmarkingový sběr dat

Údaje pro porovnávání jsou takové, které se v organizaci dobře vyhledávají, jsou lehce dostupné a uznávané. Výběr údajů se také liší tím, zda chceme benchmarking používat k zlepšení kvality nebo produktivity. V našem případě se zaměříme na zlepšení kvality vzdělání. Z dodaného excelovského souboru, který byl používán pro minulé benchmarkingové potřeby analýzy jsem vyspala tyto sbíraná data, ze kterých dále definuji ukazatele pro srovnání:

- objem financí na doplňkové činnosti
- objem financí vynaložených na výzkum projektů
- objem financí z MŠMT
- objem investičních prostředků kromě stavebních investic
- počet asistentů a lektorů
- počet článků v impaktovaném časopise – anglicky

- počet článků v impaktovaném časopise – česky, slovensky,...
- počet článků v neimpaktovaném časopise, recenzovaném – anglicky
- počet článků v neimpaktovaném časopise, recenzovaném – česky, slovensky,...
- počet docentů
- počet kapitol v odborné knize – anglicky
- počet kapitol v odborné knize – česky, slovensky, ...
- počet měsíců v rámci zahraničních studijních pobytů studentů bakalářského, magisterského a doktorského studia
- počet národních patentů
- počet odborných asistentů
- počet odborných knih – anglicky
- počet odborných knih – česky, slovensky,...
- počet ostatních pracovníků
- počet pedagogů
- počet poloprovozů a ověřených technologií
- počet přihlášených patentů a užitých vzorů
- počet přijatých studentů
- počet příspěvků ve sborníku úplný – anglicky
- počet příspěvků ve sborníku úplný – česky, slovensky,...
- počet profesorů
- počet realizovaných projektů
- počet realizovaných zahraničních studijních pobytů
- počet studentů
- počet studentů bakalářského studia
- počet studentů bakalářského studia připuštěných ke státní zkoušce
- počet studentů bakalářského studia, kteří úspěšně absolvovali státní zkoušku v prvním termínu
- počet studentů bakalářského studia, kteří úspěšně absolvovali státní zkoušku v druhém termínu
- počet studentů doktorského studia
- počet studentů magisterského studia

- počet studentů, kteří ukončili druhý ročník bakalářského studia
- počet studentů, kteří ukončili první ročník bakalářského studia
- počet studentů, kteří ukončili třetí ročník bakalářského studia
- počet světových a evropských patentů
- počet užitečných vzorů
- počet vědecko-výzkumných pracovníků
- počet výstupů VaV
- věk jednotlivých skupin pedagogů

## 1.6 Ukazatelé a jejich definice

V minulém bodě jsme definovali data, která budeme sbírat, která se tedy budou zadávat do vstupních formulářů pro benchmarking. Nyní si vytvoříme ukazatele, kteří budou působit na výstupu z informačního systému a díky nim budou moci uživatelé tohoto systému zlepšovat rozvoj daných vysokých škol. Aby byl smysl každého ukazatele jasný, uvádím zde i postup jejich výpočtu: [Šlosarčík M., 2009]

- *objem financí doplňkových činností na přepočteného pedagoga* = suma financí vynaložených na doplňkovou činnost / přepočtený počet pedagogů
- *objem financí výzkumných projektů bez doplňkové činnosti na přepočteného profesora, docenta, vědecko-výzkumného pracovníka* = suma financí vynaložených na výzkumné projekty mimo doplňkovou činnost / (součet přepočtených profesorů, docentů a vědecko-výzkumných pracovníků)
- *počet přihlášek patentů a užitečných vzorů*
- *počet studentů na přepočteného pedagoga* = počet studentů / přepočtený počet pedagogů
- *počet studentů Ph.D na přepočteného docenta a profesora* = počet studentů Ph.D / přepočtený počet docentů a profesorů
- *počet studijních měsíců výjezdů na zahraniční studijní pobyt* = počet studijních měsíců v zahraničí / počet realizovaných studijních pobytů
- *přepočtené výstupy vědy a výzkumu na pedagoga* = součet bodového hodnocení za jednotlivé přiznané výstupy vědy a výzkumu / počet pedagogů

- *přepočtený počet pedagogů* = součet pedagogů pracujících na plný úvazek a pedagogů pracujících na částečný úvazek (př. 1 pedagog na plný úvazek a 2 pedagogozi na ½ úvazek =  $1 + 2 * 0,5 = 2$ ), stejně tak lze vypočítat přepočtený počet docentů, profesorů, asistentů,....
- *procento docentů a profesorů v přepočtených pedagogích* = součet přepočteného počtu docentů a profesorů / přepočtený počet pedagogů \* 100, v [%]
- *úspěšnost studentů u státních závěrečných zkoušek z bakalářského studia v prvním termínu* = počet úspěšných studentů u závěrečných zkoušek / počet studentů připuštěných k závěrečným zkouškám \* 100, v [%]
- *úspěšnost studentů u státních závěrečných zkoušek z bakalářského studia v druhém termínu* = počet úspěšných studentů u závěrečných zkoušek / počet studentů připuštěných k druhému termínu státních závěrečných zkoušek \* 100, v [%]
- *úspěšnost studentů v druhém ročníku bakalářského studia* – více výpočet úspěšnosti v 1. ročníku bakalářského studia
- *úspěšnost studentů v prvním ročníku bakalářského studia* = počet studentů, kteří ukončili první ročník bakalářského studia / počet přijatých studentů \* 100, v [%]
- *úspěšnost studentů v třetím ročníku bakalářského studia* – více výpočet úspěšnosti v 1. ročníku bakalářského studia
- *věkový průměr docentů* – více výpočet pro věkový průměr profesorů
- *věkový průměr odborných asistentů* – více výpočet pro věkový průměr profesorů
- *věkový průměr profesorů* = součet věků profesorů / počet profesorů
- *výše investičních prostředků na přepočteného pedagoga* = suma investic / přepočtený počet pedagogů
- *výše investičních prostředků z MŠMT na přepočteného pedagoga* = suma vynaložených investic z MŠMT / přepočtený počet pedagogů

## 2 Informační systém

V této kapitole si odpovíme na základní otázky, co je informační systém, jak pracuje, jaké jsou jeho výhody, seznámíme se s postupy návrhu informačních systémů a se současnými databázovými systémy.

### 2.1 Pojem informační systém

Informační systém je systém sběru, uchovávání, analýzy a prezentace dat určený pro poskytování informací mnoha uživatelům různých profesí. Pro informační systém (IS) platí několik faktů:

- Může být podporován počítačem
- Musí disponovat prostředky sběru, kontroly a uchování dat.
- Jsou vyjasněné vztahy mezi informacemi a daty. Informace jsou jen ta data, která dokážeme využít, přiřadit jim význam či smysl. Při návrhu IS nutno umožnit získávání odlišných informací pro různé zaměstnance – skladník, ředitel atd.
- Informační systém ovlivňuje pracovní procesy i organizační struktura podniků.
- Informační systém je vždy společným dílem dodavatele a zákazníka.

Další důležitou otázkou je, jak lze informační systém získat. Máme v dvě základní možnosti. Můžeme informační systém vyvíjet od začátku na míru zadaným požadavkům nebo můžeme informační systém koupit a přizpůsobit našim potřebám = customizace.

### 2.2 Výhody a nevýhody customizovaného IS

- (+) Menší nebezpečí, že dodavatel opustí trh, neboť customizovaný IS bývá obvykle podporován větší firmou.
- (+) Obsahuje know-how mnoha instalací, dodavatel většinou poskytuje přesné postupy pro zjišťování požadavku, instalaci, školení koncových uživatelů a ožívání systému na místě.
- (+) Ověřeno na více instalacích, tudíž bychom měli být ušetřeni chyb prvotin a přebírat tak kladné zkušenosti.
- (+) Úspora nákladů na vývoj a především údržbu, což bývá zvláště u nákladů více než nezanedbatelná položka.

- (-) Neodpovídá přesně potřebám. To obvykle znamená menší účinnost a také vyšší náklady na reorganizace, které by jinak nemusely být nutné.
- (-) Stejný systém může být upravován i pro naši konkurenci, která tak má k dispozici stejné funkce a výstupy námi používaného systému, takže neposkytuje podstatnou výhodu před konkurencí.
- (-) Vyšší nebezpečí, že je IS založen na zastaralých technologiích.
- (-) U cizích systémů nedostatečná lokalizace – potíže s českou legislativou.
- (-) Obtíže s integrací produktů třetích stran a existujících aplikací.
- (+/-) Vyšší nabídka funkcí, které však nemusí být vždy potřebné a pak zbytečně zvyšují nároky na obsluhu systému a také na hardware. [Pokorný Martin, 2004]

## 2.3 Proces návrhu systému

Při vývoji IS musíme projít určitou posloupnost činností spojenou s analýzou a návrhem. Tyto činnosti bychom neměli brát na lehkou váhu, jelikož by se nám při jejich zanedbání později (při fyzickém vývoji) mohl naskytnout problém, který bychom jen stěží řešili, celá tvorba IS by pak zabrala mnohem více času a ve výsledku by IS nemusel zcela perfektně pracovat tak jak jsme si představovali na počátku. Proto zde uvádím seznam základních kroků analýzy a návrhu IS: [Pokorný Martin, 2004]

**Tab. 2 Základní kroky analýzy a návrhu ISS**

<b>Krok</b>	<b>Popis</b>
<b>1</b>	Definice parametrů systému - přesné vymezení cílů systému, co přesně má vlastně dělat a jak toho chceme dosáhnout
<b>2</b>	Definice pracovních procesů - konkrétní úkony (operace), které bude uživatel od vytvářeného systému očekávat
<b>3</b>	Sestavení myšlenkového datového modelu - definice datových struktur pro vlastní uložení dat a využití dat v celém systému
<b>4</b>	Příprava databázového schématu - převedení myšlenkového datového modelu do konkrétních fyzických pojmů
<b>5</b>	Návrh uživatelského rozhraní

Stanovení cílů projektu (SCP - projektový záměr) má obvykle následující strukturu:

1. Název projektu
2. Shrnutí cílů (formulace problému) obsahuje formulaci celkového úkolu systému formou srozumitelnou i nečlenům týmu. Tato část má být stručná a vystihnout podstatu.
3. Perspektivy realizovaného systému (doba života, možnosti předání dalším uživatelům).
4. Vymezení uživatelů (kdo, kdy a za jakých okolností bude systém využívat, případně pro koho systém není určen).
5. Seznam nejdůležitějších funkcí spolu se stručnými popisy funkcí. Popis je formulován z hlediska uživatele.
6. Zásady pro dokumentování.
7. Vazby na jiné projekty a systémy.
8. Rámcové požadavky na hardware (konfigurace, spolehlivost. ...) a požadavky na efektivnost zpracování.
9. Metody ochrany dat, žádoucí způsoby využívání dodávaného softwaru.
10. Požadavky na spolehlivost systému jako celku (doba mezi chybami, vzpamatování po chybě, ochrana dat, funkce nutné pro detekci chyb).
11. Předpokládané termíny realizace a náklady na realizaci.
12. Způsob předání.
13. Perspektivy realizovaného systému, jeho další rozvoj a zajištění údržby, pravidla šíření.

**Tab. 3Projektový záměr**

<b>SCP – PROJEKTOVÝ ZÁMĚR PRO IS BENCHMARKINGU</b>	
<b>Krok</b>	<b>Popis</b>
<b>Název projektu</b>	Benchmarking vysokých škol
<b>Cíl</b>	Vytvořit informační systém pro porovnávání různých kritérií vysokých škol
<b>Perspektivy</b>	Online srovnání bez potřeby domluvy předání údajů apod.

<b>Uživatelé</b>	Registrovaní uživatelé
<b>Nejdůležitější funkce</b>	Vyhledávání, přidávání ukazatelů, srovnání, vkládání dat
<b>Dokumentování</b>	-
<b>Vazby na jiné projekty</b>	Žádné
<b>Hardwarové a softwarové požadavky</b>	Standartní uživatelské pc s přístupem na internet
<b>Ochrana dat</b>	Zabezpečení v rámci serveru, každý uživatel má své heslo
<b>Spolehlivost systému</b>	-
<b>Termín a náklady na realizaci</b>	Přibližný termín spuštění duben/květen 2010, náklady 0 Kč
<b>Způsob předání</b>	Databáze sql, aplikační prostředí ASP.net, CD-ROM
<b>Další rozvoj</b>	Zálohování 1x ročně k určitému datu.

## 2.4 Databáze a databázové systémy

Databází rozumíme pouze ono skladiště dat, ve kterých jsou data uložena a zpracovávána nezávisle na aplikačních programech. Pro samotný přístup k datům uloženým v databázi používáme speciální software. Nazývá se anglicky Database Management System (DBMS) nebo česky Systém řízení báze dat (SŘBD). SŘBD je softwarové vybavení, které zajišťuje práci s databází, tzn. Tvoří rozhraní mezi aplikačními programy a uloženými daty. Fyzická struktura uložených údajů potom nemusí být aplikačnímu programu, ani uživateli, vůbec známa. Historicky se vyvinuly tři hlavní databázové modely, a to síťový, hierarchický a relační.

### 2.4.1 Hierarchický model

Nejstarší z uvedených je hierarchické modelování databází. Toto pojetí pochází z reálného uspořádání světa. Jako příklad si můžeme vzít třeba model organizace moci, rozklad výrobků na součástky, strom adresářů aj. Pro hierarchické modelování je typická práce se stromy, kdy ve stromu jsou realizovány vztahy 1:N.



### 2.4.2 Síťový model

Variací hierarchického modelu je síťový model databáze. V síťovém modelování je možné vyjadřovat vedle vztahů 1:N i vztahy M:N. Fyzická realizace síťového modelu je ale náročná a aktualizace obvykle komplikovaná. Základní výhodou hierarchického a síťového modelu je efektivnost zpracování, tj. rychlost přístupu k datovým záznamům. Na druhé straně mezi nevýhody patří to, že je nesnadné jednou nadefinované stromy a vazby mezi nimi měnit. Nejsou uzpůsobeny pro dotazy.

### 2.4.3 Relační model

Nejpropracovanější je relační model databáze, který byl vyvinut doktorem E. F. Coddem už v šedesátých letech minulého století. Relační model definuje způsob, jakým je možné reprezentovat strukturu dat, způsoby jejich ochrany a operace, které můžeme nad daty provádět. Relační databáze je sestavená z řady tabulek, jejichž sloupce jsou vázány na sloupce v jiných tabulkách. Takto propojená datová pole jsou na sobě určitým způsobem závislá. Jejich vztahy jsou založeny na klíčových hodnotách uložených v příslušných sloupcích. U relačních databází je základní výhodou relativně snadná modifikace a propojování tabulek a s nimi spojená možnost dotazů. V současnosti je dále využíván objektový a objektově-relační přístup k databázím.

### 2.4.4 Dotazovací jazyk

S relačními databázovými systémy se samozřejmě začaly vyvíjet a postupně zdokonalovat dotazovací jazyky. V praxi se můžeme setkat se dvěma koncepty dotazovacích jazyků:

- koncept QBE (Query By Example)
- koncept SQL (Structured Query Language)

Při práci s QBE jde spíše o jednoduchý zápis dotazu do navrženého schématu (formuláře) než o "programování". Koncept QBE tak umožňuje rychle vytvářet dotazy i běžným uživatelům, nejen odborníkům. Původní vznik konceptu QBE byl spojen s firmou IBM už v 70. letech. V letech 1974 až 1975 probíhal ve firmě IBM výzkum týkající se možnosti využití relačních databází. Pro tento projekt vznikl jazyk SEQUEL (Structured English Query Language), který měl co nejvíce napodobovat běžný jazyk (angličtinu). IBM svůj jazyk dále vylepšoval a začaly vznikat i databázové platformy dalších firem. V těchto systémech se používaly různé verze jazyka SEQUEL, který se později přejmenoval na SQL. Jazyk SQL byl postupně přijat jako standard různými výrobci databázových aplikací a stal se tak spojovacím článkem mezi různými systémy. V koncepci klient/server se dotazy specifikují na straně klienta, odesílají na stranu serveru, který dotaz v jazyce SQL realizuje a výsledek pošle zpět uživateli na straně klienta. Většina významných databázových platforem dnes jazyk SQL podporuje. Vznikající význam relačních databází si vyžádal nutnost

standardizace. A tak v roce 1986 byl Americkým standardizačním institutem (ANSI) přijat nový standard SQL-86. SQL-86 měl však některé nedostatky a tak byl v roce 1992 přijat standard SQL-92 (nebo také SQL2). V současnosti je v platnosti norma SQL3, která reaguje na poslední trendy vývoje v databázových technologiích (např. využívání objektů apod.)

Jazyk SQL můžeme částečně využít pro samotný vývoj databázových aplikací, ale jinak slouží především jako dotazovací jazyk pro práci s údaji v relační databázi. Příkladem vyspělých SŘBD jsou produkty Oracle, Microsoft Access, Microsoft SQL Server, MySQL, Microsoft Visual FoxPro a další. [Pokorný Martin, 2004]

## 2.5 Současné databázové systémy

Jak už bylo v minulé kapitole naznačeno budeme se nyní zabývat popisem současných, nejznámějších databázových systémů. Úkolem této kapitoly je poznat jednotlivé databázové systémy abychom se mohli rozhodnout, který systém budeme používat v naší aplikaci.

### 2.5.1 MySQL

- multiplatformní databázový systém
- je považován za průkopníka dvojího licencování – k dispozici existuje bezplatná licence, ale i komerční placená licence
- komunikace probíhá pomocí jazyk SQL
- snadná implementovatelnost (Linux, Microsoft Windos,...)
- velmi oblíbená kombinace je MySQL, PHP a Apache jako základní software webového serveru
- volně šířitelný software = open source: <http://www.mysql.com/downloads/>
- systém MySQL je využitelný v C, C++, Java, Perl, PHP, Python, Tcl, Visual Basic, .NET
- optimalizován na rychlost, a to i za cenu některých zjednodušení (jednoduché zálohování, donedávna nepodpora pohledů, triggerů a uložených procedur.

### 2.5.2 Oracle

- moderní multiplatformní databázový systém
- pokročilé možnosti zpracování dat
- vysoký výkon
- standardní dotazovací jazyk SQL – norma SQL92
- dále podporuje objektové databáze, databáze XML

### 2.5.3 Microsoft Access

- součástí balíčku Microsoft Office
- nástroj na správu relačních databází
- kombinuje relační Microsoft Jet Database Engine s grafickým uživatelským rozhraním
- umí přistupovat k datům z Access/Jet, MS SQL Serveru, Oracle

### 2.5.4 FoxPro

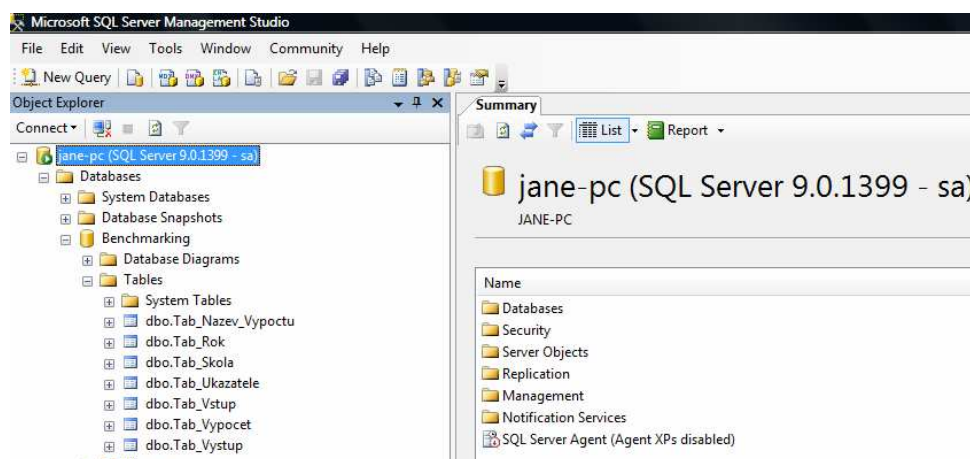
- systém řízení databáze pro prostředí operačního systému MS-DOS
- ukládání dat ve formátu DBF
- později také pro Windows, Unix a Macintosh
- nyní současná moderní podoba Microsoft Visual FoxPro 9.0 (je vývojový prostředek na databázová řešení všech rozměrů. Silnými stránkami jsou databázové jádro nad lokálním kurzorem, těsná vazba mezi programovacím jazykem a daty a celá řada efektivních zabudovaných nástrojů)

### 2.5.5 Microsoft SQL Server

- relační databázový systém
- hlavní dotazovací jazyky: SQL a T-SQL
- slouží pro vytváření, konfiguraci a údržbu databází
- zlepšuje dostupnost, výkon, spolehlivost, bezpečnost a možnosti programování
- instalovatelný na různé hardwarové platformy
- 6 verzí systému SQL Serveru (Mobile Edition – pro kompaktní databáze do 2MB, pro mobilní telefony; Express Edition – bezplatná verze s omezením kapacity; Workgroup Edition – databáze bez omezení velikosti, nabízí většinu funkcí, pro malé organizace; Standard Edition – vysoce dostupná platforma, u středně velkých podniků; Enterprise Edition – nejrobustnější verze, nejlépe se hodí pro kritické podnikové pracovní úlohy; Developer Edition – obsahuje všechny funkce a možnosti verze Enterprise, je přizpůsobena individuálním vývojářům)

### 3 Návrh a ověření datové struktury

Datovou strukturu informačního systému pro podporu realizace benchmarkingu jsem převzala od Martina Šlosarčíka [Šlosarčík M.,2009], jelikož se mi zdála dobře koncipována. Namísto Microsoft Access jsem však k tvorbě databáze použila SQL server 2008. Na obrázcích 2 - 6 uvádím použité tabulky včetně seznamů použitých argumentů a jejich datových typů. Také zde uvádím na ukázkou několik náhledů tabulek naplněných daty.




Obr. 2 Obrazovka programu MS SQL Server 2005 včetně seznamu tabulek

Tabulka škola obsahuje položky různých datových typů. Jako primární klíč je zde uvedeno identifikační číslo, které je datového typu integer s přírůstkovou hodnotou rovnou jedné, tzn. při zadávání konkrétních údajů, bude pro nový údaj automaticky generováno nové pořadové číslo. Ostatní položky jsou datového typu nvarchar se zadanou maximální hodnotou délky textu nebo typu datetime pro vložení datumu. Také zde uvádím nutnost zadání hodnoty, viz. sloupec Allow Nulls.

Table - dbo.Tab_Skola			
Summary			
	Column Name	Data Type	Allow Nulls
🔑	ID	int	<input type="checkbox"/>
	Nazev_Skoly	nvarchar(100)	<input type="checkbox"/>
	Adresa_Skoly	nvarchar(50)	<input type="checkbox"/>
	Fakulta	nvarchar(50)	<input checked="" type="checkbox"/>
	Katedra	nvarchar(50)	<input checked="" type="checkbox"/>
	Zkratka	nvarchar(50)	<input type="checkbox"/>
	Kontaktni_Osoba	nvarchar(50)	<input type="checkbox"/>
	Email	nvarchar(30)	<input checked="" type="checkbox"/>
	Telefon	int	<input checked="" type="checkbox"/>
	Poznamka	nvarchar(50)	<input checked="" type="checkbox"/>
	Jmeno_Uzivatele	nvarchar(20)	<input type="checkbox"/>
	Heslo_Uzivatele	nvarchar(20)	<input type="checkbox"/>
	Datum_Registrace	datetime	<input type="checkbox"/>



Obr. 3 Tabulka škola v návrhovém prostředí

Tabulka ukazatele slouží k přidávání nových ukazatelů benchmarkingu. Obsahuje pouze identifikační číslo ID, které je opět datového typu integer a vedeno jako automatické číslo s přírůstkem jedna. Druhou položkou je název ukazatele, který je datového typu nvarchar a je omezen na 100 znakovou délku.

Table - dbo.Tab_Ukazatele		Table - dbo.Tab_Skola	Summary
Column Name	Data Type	Allow Nulls	
 ID	int	<input type="checkbox"/>	
Nazev_ukazatele	nvarchar(100)	<input type="checkbox"/>	
		<input type="checkbox"/>	


Obr. 4 Tabulka ukazatelů v návrhovém zobrazení

Tabulka vstup bude sloužit při zadávání hodnot jednotlivých ukazatelů. Všechny položky jsou datového typu integer a většina z nich odkazuje na jinou tabulku (jsou primárními klíči jiných tabulek).

Table - dbo.Tab_Vstup		Summary
Column Name	Data Type	Allow Nulls
 ID	int	<input type="checkbox"/>
ID_Rok	int	<input type="checkbox"/>
ID_Skola	int	<input type="checkbox"/>
ID_Ukazatel	int	<input type="checkbox"/>
Hodnota	int	<input type="checkbox"/>
		<input type="checkbox"/>

Obr. 5 Tabulka vstup v návrhovém zobrazení

Tato tabulka je obdobou Tabulky vstup, také zde je většina položek cizími klíči, ale bude sloužit k tvorbě výstupních grafů.

Table - dbo.Tab_Vypocet		Table - dbo.Tab_Vystup	Summary
Column Name	Data Type	Allow Nulls	
 ID	int	<input type="checkbox"/>	
ID_Rok	int	<input type="checkbox"/>	
ID_Skola	int	<input type="checkbox"/>	
ID_Ukazatel	int	<input type="checkbox"/>	
ID_Vypocet_ID_Ukazatel	int	<input type="checkbox"/>	
Hodnota	int	<input type="checkbox"/>	
		<input type="checkbox"/>	

Obr. 6 Tabulka výstup v návrhovém zobrazení

Tabulka výpočet bude sloužit k zadávání výpočtu jednotlivých ukazatelů, např. Ukazatel počet studentů na přepočteného pedagoga = počet studentů Bc. (C1) + Mgr. (C2) + Ph.D. (C3) studia / přepočtený počet pedagogů (J1). Ukazatel tedy zapíšeme jako  $(C1 + C2 + C3) / J1$ .

Table - dbo.Tab_Vypocet		Summary
Column Name	Data Type	Allow Nulls
ID	int	<input type="checkbox"/>
Nazev_Vypoctu	int	<input type="checkbox"/>
Rok	int	<input type="checkbox"/>
C1	int	<input checked="" type="checkbox"/>
C2	int	<input checked="" type="checkbox"/>
C3	int	<input checked="" type="checkbox"/>
C4	int	<input checked="" type="checkbox"/>
J1	int	<input checked="" type="checkbox"/>
J2	int	<input checked="" type="checkbox"/>
J3	int	<input checked="" type="checkbox"/>
J4	int	<input checked="" type="checkbox"/>
J5	int	<input checked="" type="checkbox"/>
J6	int	<input checked="" type="checkbox"/>

Obr. 7 Tabulka výpočet v návrhovém zobrazení

Pro představu naplnění tabulek daty, je zde zobrazena Tabulka škola i s vzorovými daty. Všechny tabulky jsem naplnila, abych ověřila funkčnost databáze a dále s nimi budu pracovat v dotazech, formulářích, atd.

Table - dbo.Tab_Skola		Table - dbo.Tab_Vypocet	Summary			
	ID	Nazev_Skoly	Adresa_Skoly	Fakulta	Katedra	Zkratka
▶	1	Vysoká škola Báňská Technic...	17.listopadu,Ostrava Poruba	Strojní	352	352-FS-VŠB Ostrava
	2	Vysoké učení technické Brno	1.máje,Brno střed	Strojní	Info	KI-FS-VUT Brno
	4	Technická univerzita Plzeň	Plzeňská 356,Plzeň	Strojní	Technologie	KT-FS-TU Plzeň
	5	TnUAD Trnava	835, Trnava	Strojní	Technologie	KT-FS-TnUAD Trnava
	7	SJF ŽU Žilina	6.ledna,Žilina	Strojní	Bleble	KB-FS-SJFŽU Žilina
	8	ČVUT v Praze	Staroměstská 888,Praha-centrum	Strojní	NULL	FS-ČVUT Praha
	9	MTF STU Bratislava	Okrová 458, Bratislava	Metalurgie	Kovy	KK-FM-STU Bratislava
	11	FS TU Liberec	3.října 444,Liberec	Strojní	NULL	FS-TU Liberec
	12	SJF STU Bratislava	Bratislava	Strojní	458	458-FS-STU Bratislava

Obr. 8 Naplněná tabulka škola

Kontaktní_Osoba	Email	Telefon	Poznamka	Jmeno_Uzivatele	Heslo_Uzivatele	Datum_Registr...
Radim Farana	radim.farana@vsb.cz	777111222	NULL	RF	RF	12.3.2004 0:00:00
Alfons Mucha	muska@seznam.cz	NULL	volat po 18.h	AM	AM	30.5.2006 0:00:00
Josef Květák	NULL	558611116	NULL	JK	JK	8.12.2004 0:00:00
Jožo Ráž	jozka@centrum.sk	555555555	NULL	JR	JR	1.1.2007 0:00:00
Jitka Pokorná	jituska@zilina.sk	654111585	NULL	JP	JP	3.1.2007 0:00:00
Karel Čtvrtý	NULL	NULL	NULL	KČ	KČ	31.1.2008 0:00:00
Karel Čapek	NULL	558666111	NULL	KČ	KČ	16.2.2007 0:00:00

Obr. 9 Pokračování tabulky škola naplněné daty

Před návrhem a tvorbou aplikační vrstvy jsem vytvořila i několik dotazů. Zde je zápis dotazu v sql jazyce. Z tabulky škola vybíráme 5sloupečků (název, zkratka, kontaktní osoba, email a telefon).

```
jane-pc.Benchmark...kolyKontakty.sql Summary
USE Benchmarking;
GO
SELECT Nazev_Skoly, Zkratka, Kontaktni_Osoba, Email, Telefon
FROM dbo.Tab_Skola;
GO
```

Obr. 10 Seznam škol s kontaktními osobami

Výstupem tohoto dotazu je výpis škol s kontaktními osobami.

	Nazev_Skoly	Zkratka	Kontaktni_Osoba	Email	Telefon
1	Vysoká škola Báňská Technická univerzita Ostrava	352-FS-VŠB Ostrava	Radim Farana	radim.farana@vsb.cz	777111222
2	Vysoké učení technické Bmo	KI-FS-VUT Bmo	Alfons Mucha	muska@seznam.cz	NULL
3	Technická univerzita Plzeň	KT-FS-TU Plzeň	Josef Květák	NULL	558611116
4	TnUAD Tmava	KT-FS-TnUAD Tmava	Jožo Ráž	jozka@centrum.sk	555555555
5	SjF ŽU Žilina	KB-FS-SjFŽU Žilina	Jitka Pokomá	jituska@zilina.sk	654111585
6	ČVUT v Praze	FS-ČVUT Praha	Karel Čtvrtý	NULL	NULL
7	MTF STU Bratislava	KK-FM-STU Bratislava	Karel Čapek	NULL	55866111
8	FS TU Liberec	FS-TU Liberec	Jan Novák	novak@centrum.cz	444555888
9	SjF STU Bratislava	458-FS-STU Bratislava	Ota Pepř	NULL	NULL

Obr. 11 Seznam škol

Dalším dotazem je seznam názvu škol, jejich identifikačních čísel včetně roků.

	Rok	ID školy	Název školy
4	2004	4	Technická univerzita Plzeň
5	2004	4	Technická univerzita Plzeň
6	2004	4	Technická univerzita Plzeň
7	2005	4	Technická univerzita Plzeň
8	2005	4	Technická univerzita Plzeň
9	2005	4	Technická univerzita Plzeň
10	2005	4	Technická univerzita Plzeň
11	2005	4	Technická univerzita Plzeň
12	2007	4	Technická univerzita Plzeň
13	2007	4	Technická univerzita Plzeň
14	2007	4	Technická univerzita Plzeň
15	2007	4	Technická univerzita Plzeň
16	2007	4	Technická univerzita Plzeň
17	2004	1	Vysoká škola Báňská Technická univerzita Ostrava
18	2004	1	Vysoká škola Báňská Technická univerzita Ostrava
19	2004	1	Vysoká škola Báňská Technická univerzita Ostrava
20	2004	1	Vysoká škola Báňská Technická univerzita Ostrava
21	2004	1	Vysoká škola Báňská Technická univerzita Ostrava

Obr. 12 Seznam škol v jednotlivých letech

```

jane-pc.Benchmark...ot_SkolyRoky.sql Table - dbo.Tab_Rok Table - dbo.Tab_Skola Table - dbo.Tab_Vstup Summary
USE Benchmarking;
GO
SELECT Tab_Rok.Rok, Tab_Skola.Id as 'ID školy', Tab_Skola.Nazev_Skoly as 'Název školy'
FROM Tab_Rok RIGHT JOIN
(Tab_Skola INNER JOIN Tab_Vstup ON Tab_Skola.ID = Tab_Vstup.Id_Skola) ON Tab_Rok.ID = Tab_Vstup.Id_rok;
GO

```

Obr. 13 Seznam škol v jednotlivých letech v sql jazyce

Posledním dotazem je dotaz ze dvou tabulek. Z tabulek škola a výstup vybíráme sloupce id, název školy, název ukazatele a hodnota.

```

jane-pc.Benchmark...kolyKontakty.sql Summary
USE Benchmarking;
GO
SELECT dbo.Tab_Vystup.ID, dbo.Tab_Skola.Nazev_Skoly, dbo.Tab_Ukazatele.Nazev_ukazatele,
dbo.Tab_Vystup.Hodnota
FROM dbo.Tab_Vystup, dbo.Tab_Skola, dbo.Tab_Ukazatele
WHERE dbo.Tab_Skola.ID=dbo.Tab_Vystup.ID_Skola and
dbo.Tab_Ukazatele.ID=dbo.Tab_Vystup.ID_Ukazatel;
GO

```

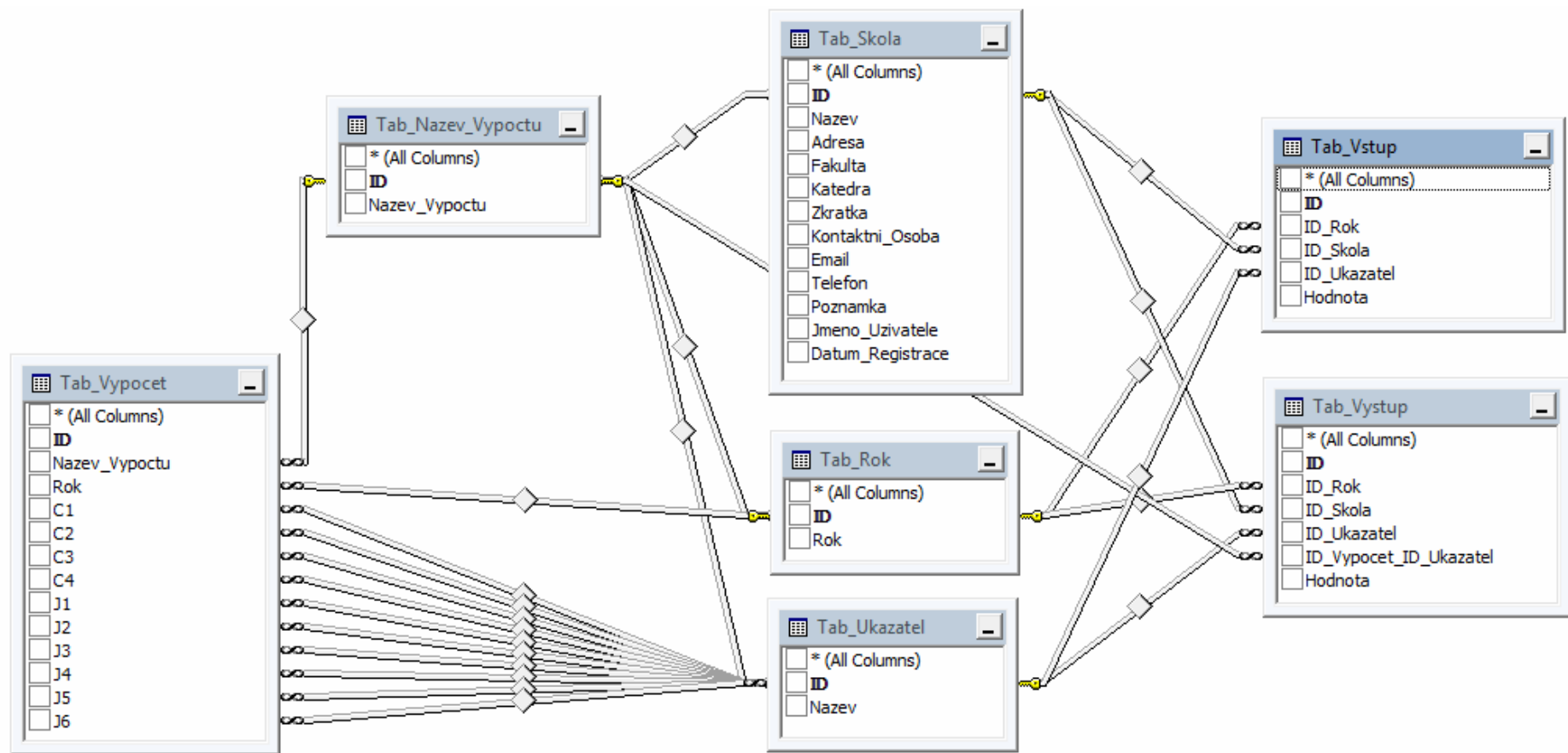
Obr. 14 Hodnoty výstupů ukazatelů pro jednotlivé školy - sql zápis

Výstupem jsou hodnoty jednotlivých škol pro jednotlivé ukazatele. Mohli bychom zde ještě přidat rok, abychom věděli rok zadání.

Results		Messages		
	ID	Nazev_Skoly	Nazev_ukazatele	Hodnota
1	3	Vysoké učení technické Bmo	Počet úspěšných studentů 1.ročníku bakalářského ...	15
2	4	Vysoká škola Báňská Technická univerzita Ostrava	Počet studentů doktorského studia k 31.10.	10
3	5	Technická univerzita Plzeň	Počet úspěšných studentů 3.ročníku bakalářského ...	15368
4	6	Technická univerzita Plzeň	Počet studentů doktorského studia k 31.10.	147

Obr. 15 Hodnoty výstupů ukazatelů pro jednotlivé školy





Obr. 16 Datová struktura databáze


## 4 Návrh struktury aplikační vrstvy IS

Než začnu tvořit aplikační vrstvu vytvářeného informačního systému, navrhnu první její předběžnou strukturu, podle které pak budu informační systém tvořit. Po načtení webové stránky se objeví úvodní obrazovka, tzv. rozcestník webu, kde si z levé nabídky budu moci vybrat, co chci dále dělat.

WELCOME		
<b>Odhlášen</b> Přihlášení Uživatel. jméno <b>Jméno</b> Heslo <b>Heslo</b> Registrace <u>Benchmarking</u> Srovnání Zadání údajů	Úvodní text	Dnes je <b>Datum</b> Svátek slaví <b>Jméno</b> Počet registrovaných <b>Počet</b>

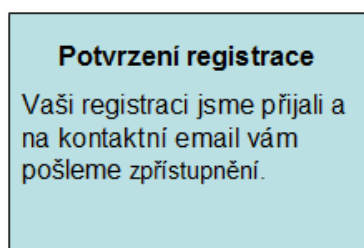
Obr. 17 Úvodní obrazovka benchmarkingu

Po zvolení nabídky registrace se otevře stránka registrace, kde se bude moci nový uživatel registrovat. Před vyplněním registračních údajů bude muset odsouhlasit pravidla benchmarkingu.

Registrace		
Pro přihlášení do tohoto systému musíte potvrdit, že souhlasíte s podmínkami <u>benchmarkingu</u> a vyplnit registrační formulář.		
<div style="border: 1px solid black; padding: 5px;"><p>Souhlasím s mlčenlivostí. Budu vkládat nesmyslná data.....</p><p>~~~~~</p><p>~~~~~</p></div>	Název školy Adresa Fakulta Kontaktní osoba Telefon Email Uživatelské jméno Uživatelské heslo	<b>Název</b> <u><b>Adresa</b></u> <u><b>Fakulta</b></u> <u><b>Kont_osoba</b></u> <u><b>Telefon</b></u> <u><b>Email</b></u> <u><b>Uzivatel_im</b></u> <u><b>Heslo</b></u>

Obr. 18 Registrační formulář

Po dokončení registrace se zobrazí dialogové okno (viz. Obr. 19), které oznámí uživateli, že v nejbližší době bude na email kontaktován o zprovoznění srovnávání a zadávání údajů pro potřeby benchmarkingu vysokých škol.



Obr. 19 Potvrzení registrace

Jednou z dalších nabídek hlavní stránky je srovnávací analýza, kde si uživatel vybere ze seznamu ukazatelů ten, který jej zajímá, dále srovnávací období, popřípadě partnera, se kterým se chce srovnat.

A light blue rectangular form with a thin black border. At the top, the title 'Srovnávací analýza' is centered in bold. To the right of the title is a small cartoon house icon with a smiling face. Below the title, there are three main sections. The first section is labeled 'Vyberte ukazatel:' followed by a text input field and a dropdown arrow. The second section is labeled 'Srovnávací období:' followed by 'od' and 'do' labels, each with a corresponding text input field. The third section is labeled 'Vyberte partnera/y:' followed by a text input field and a dropdown arrow. At the bottom center of the form is a button labeled 'Provést srovnání'.

Obr. 20 Formulář srovnávací analýzy

Po stisknutí tlačítka „Provést srovnání“ se zobrazí nová stránka, kde bude konkrétní ukazatel zobrazen. Dále je zde možnost přepnutí zobrazení mezi grafem a výstupem ve formě tabulky. Oba dva výstupy bude možno poslat na tisk, uložit dále do počítače, popřípadě se vrátit zpět na předešlý formulář.

Obr. 21 Formulář vyhodnocení

Pro srovnávací analýzu je potřeba mít vyplněné údaje pro srovnání. Tyto údaje bude možno zadat do připraveného formuláře.

Počet studentů přijatých do 1.r. Bc. Studia	548
Počet profesorů na HPP	10
Počet docentů	6
Věkový součet profesorů	712
Objem financí vědy a výzkumu	120 000
Počet zahraničních výjezdů k 31.10.	8
Počet přijatých patentů	4
Počet studentů, kteří úspěšně vykonali SZZ v 1.t.	32

Obr. 22 Formulář zadávání údajů

V případě opětovného zadání údajů bude zobrazeno dialogové okno, kde si bude moci uživatel vybrat zda chce nově zadaná data přepsat, popřípadě ponechat starší data.

Obr. 23 Přepisovací dialogové okno

## 5 Prostředí pro tvorbu aplikační vrstvy

K vytváření aplikační vrstvy jsem si vybrala program ASP.NET, se kterým jsem se setkala při studiu již dříve. ASP.NET je technologie, která konkuruje technologii PHP. K tvorbě stránek v ASP.NET se předpokládá, že znáte HTML, CSS a máte tušení jak funguje databáze, web a skripty na straně serveru. I pro napsání jednoduché aplikace se vyžadují poměrně široké znalosti .NET frameworku, objektově orientovaného programování a technologie ASP.NET samotné. Protože s touto technologií teprve začínám, nemusí být všechny mé myšlenky nejlepší. Dříve než začnu webovou aplikaci tvořit, aspoň stručně popíši výhody, nevýhody a prostředí tohoto programu.

### 5.1 Výhody a nevýhody ASP.NET

Mnoho lidí si ASP.NET spojuje se starším ASP, avšak architektura a princip je úplně jiný. ASP se mnohem více podobá PHP. Mezi nejdůležitější výhody ASP.NET patří:

- Bezplatné vývojové prostředí
- Přehledné grafické prostředí
- Jednoduché přidávání uživatelů, rolí, profile, přístupových práv
- Nastavení a konfigurace na jednom místě
- Technologie běží na .NET frameworku - máme k dispozici spoustu funkcí například pro šifrování a zabezpečení, práce se soubory, grafikou, práci s daty, atd.
- Podpora přístupu různým klientům, jako třeba mobilní telefony s omezenými možnostmi.

Jako každý program i tato technologie má své milovníky i odpůrce. Hlavními nevýhodami jsou:

- Složitost technologie
- Menší počet poskytovatelů webhostingu
- Špatná dostupnost česky psané literatury
- Nepříliš populární výstupní XHTML kód, ve kterém se může objevit nelogické odsazování, zbytečná volná místa v kódu, občas i nesmyslně zvolené značky, atd.).
- Při používání určitých komponent, je ASP.NET závislé na JavaScriptu

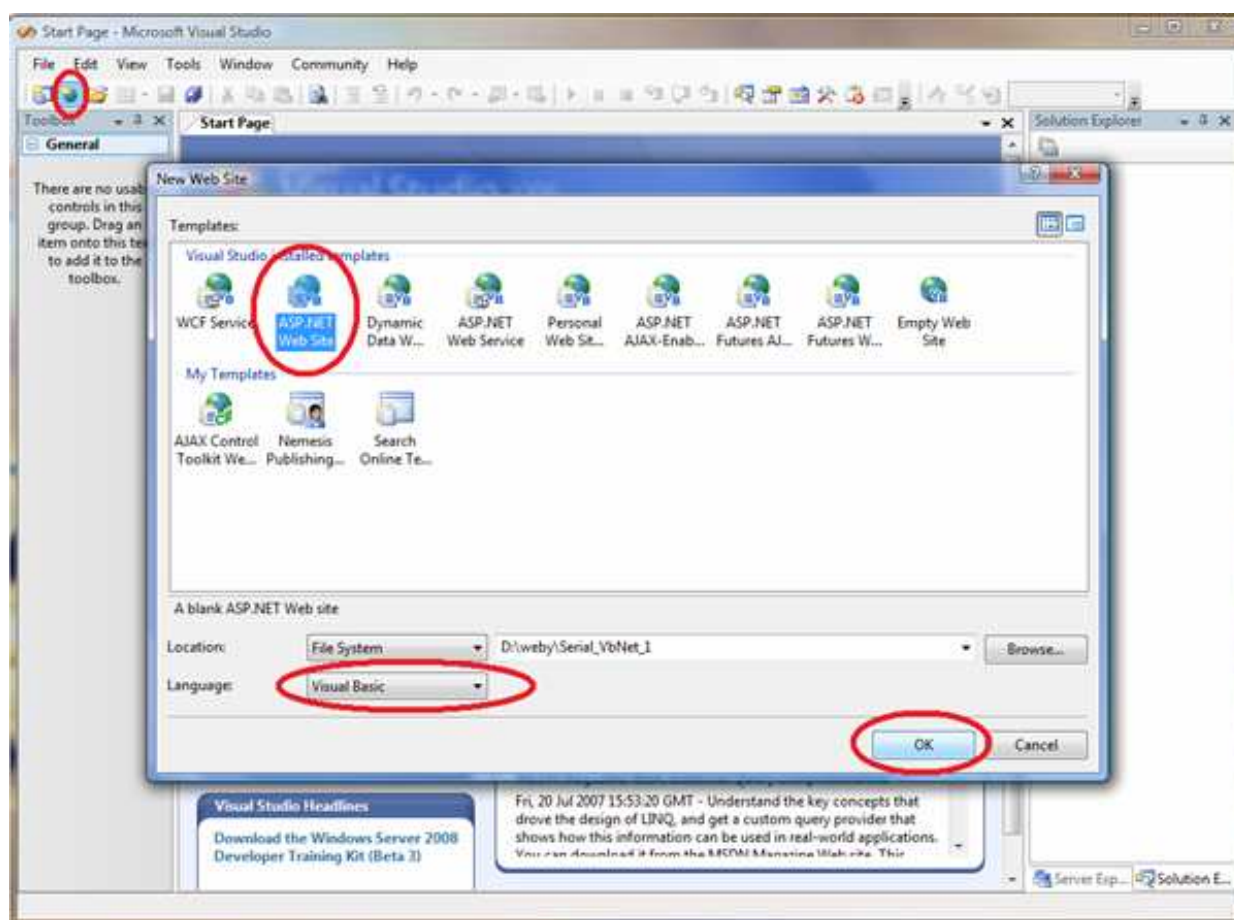
K vývoji aplikace v tomto prostředí je potřeba vývojové prostředí Visual Studio 2005 (popřípadě nová verze Visual Studio 2008), které Microsoft uvolnil k volnému použití a lze jej zdarma získat.

Tato aplikace není dostupná v češtině, proto může některým začínajícím uživatelům dělat problémy. Další produkt, který je potřeba je SQL Server 2005 a novější, ten je také volně dostupný.

## 5.2 Prostředí ASP.net

Nyní se budu věnovat krátkému úvodu do tohoto vývojového prostředí. Nebudu zde popisovat veškerou práci pouze vytvoření nové aplikace a vytvoření jednoduchého formuláře.

Pro vytvoření nové aplikace si spustíme Visual Studio, dáme přidat novou webovou aplikaci, zvolíme úložiště a používaný programový jazyk. Jsou zde dvě volby C# a Visual Basic. Při mé tvorbě budu pracovat s jazykem Visual Basic.

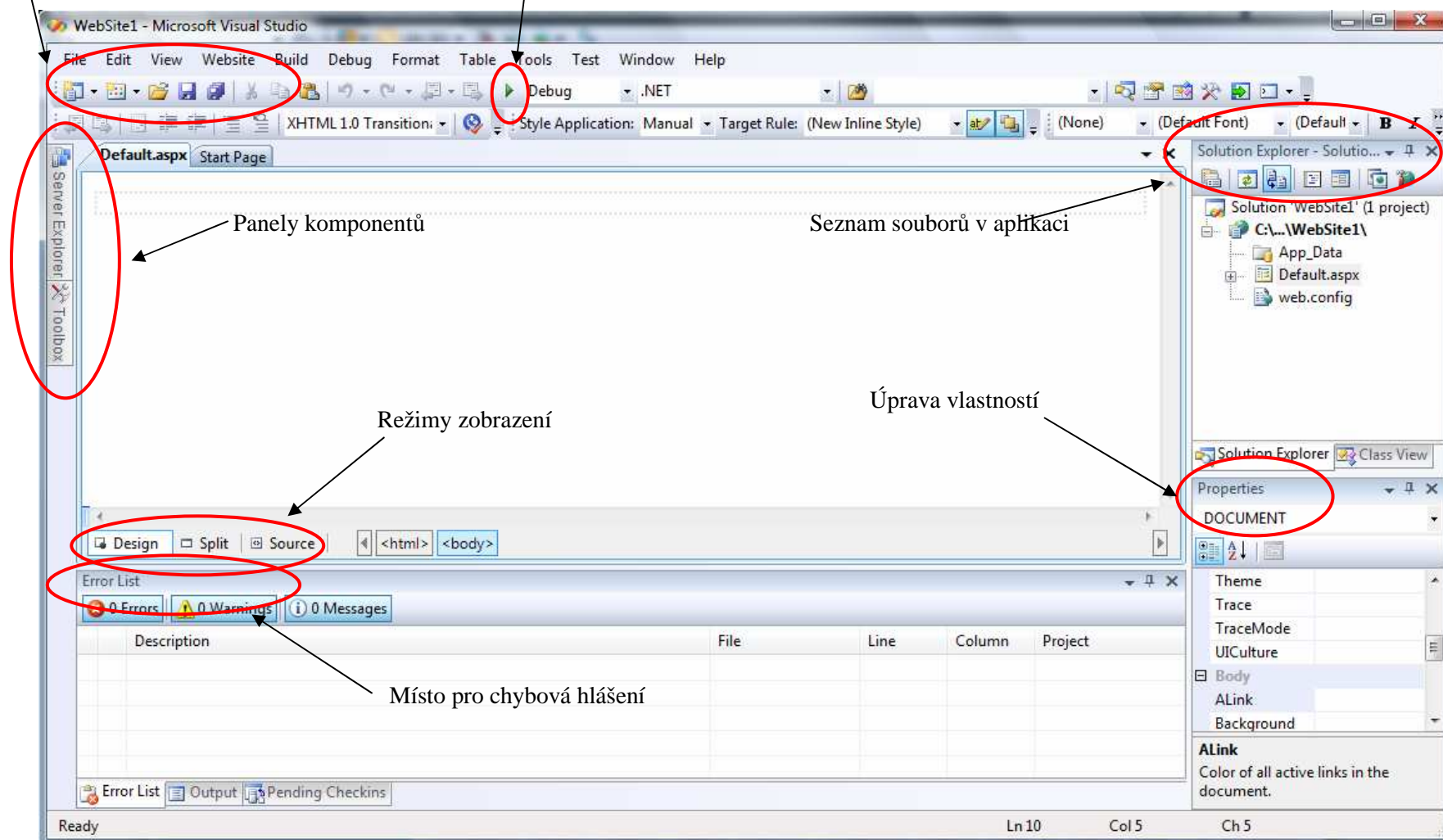


Obr. 24 Vytvoření webové aplikace

Po potvrzení se objeví nova stránka na kterou můžeme umístit komponenty. Komponenty jako třeba TextBox (textové pole), Button (tlačítko) nebo Hyperlink (odkaz) mají své vlastnosti, metody a události. S webovou stránkou lze pracovat v režimu design a source. V režimu Design se stránka zobrazuje přibližně tak, jak bude vypadat v prohlížeči, naproti tomu režim Source slouží spíše k programování, práci se zdrojovým kódem, vlastnostmi komponentů atd.

Standartní panel nástrojů

Spuštění aplikace



Obr. 25 Prostředí ASP.NET

## 6 Připojení databáze ve Visual Studiu 2008

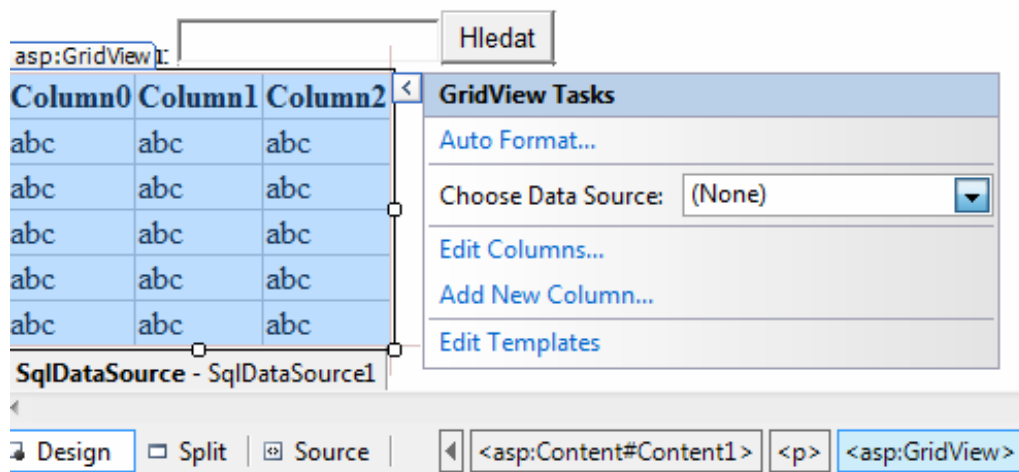
Databázi SQL Serveru připojíme do webové aplikace pomocí připojovacího řetězce. Tento řetězec se vkládá do stránky web.config do sekce `<connectionStrings>`. Zde je uveden příklad tohoto připojení:

```
<connectionStrings>

<add name="benchmarking" connectionString="DataSource=.\SQLEXPRESS;Integrated
Security=True; User Instance=True; AttachDBFilename=|DataDirectory|testDB.mdf;
Database=DBbenchmarking" providerName="System.Data.SqlClient" />
```

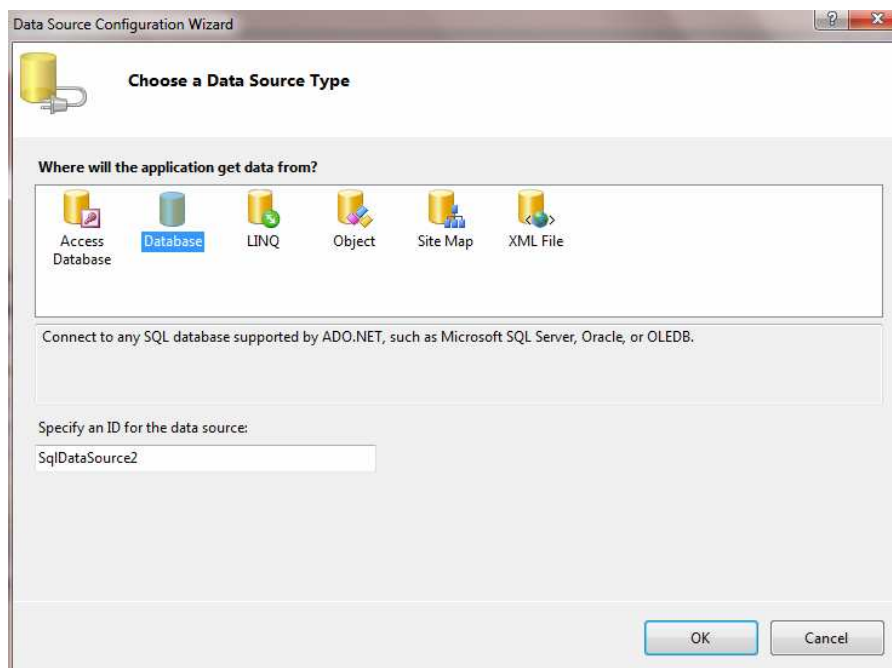
Dále použijeme metodu `Open()`, která je částí příkazu `SqlConnection`. Na konci práce s daty databázi nesmíme zapomenout uzavřít.

Další metoda jak se k databázi připojit se nachází v designovém režimu Microsoft Visual Studia, viz. Obr. 27. Vybereme zde druh databáze, název databáze, Obr. 29, příslušnou tabulku a data, viz. Obr. 30 (máme zde možnost také data řadit, seskupovat a vytvářet nad nimi dotazy) a připojení k databázi je dokončeno.

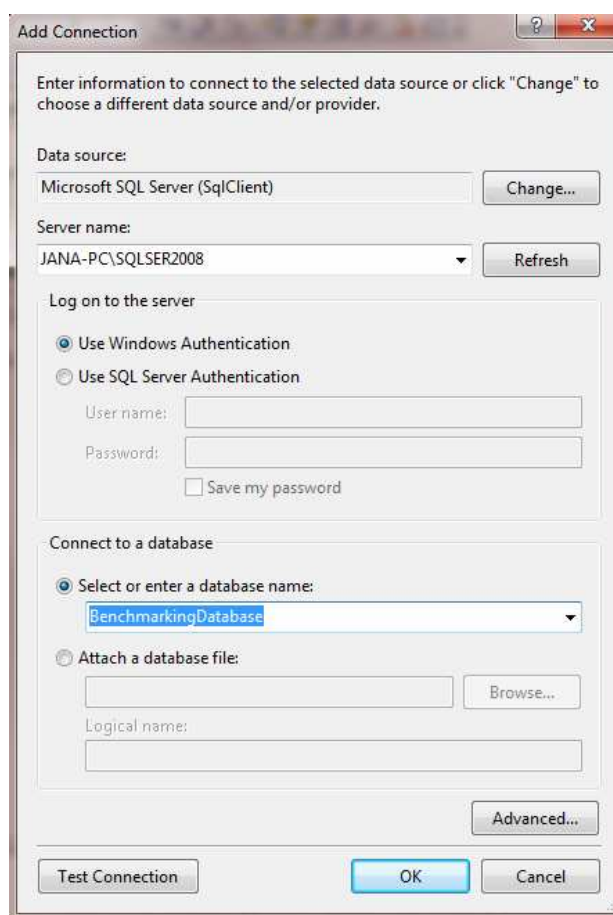


Obr. 27 Připojení databáze v design režimu





Obr. 28 Druh databáze



Obr. 29 Výběr serveru a příslušné databáze

Configure Data Source - SqlDataSource1

**Configure the Select Statement**

How would you like to retrieve data from your database?

☐ Specify a custom SQL statement or stored procedure

☒ Specify columns from a table or view

Name:  
Tab\_Skola

Columns:

<input type="checkbox"/> *	<input type="checkbox"/> Poznamka
<input type="checkbox"/> ID	<input type="checkbox"/> Jmeno_Uzivatele
<input checked="" type="checkbox"/> Nazev	<input type="checkbox"/> Heslo_Uzivatele
<input checked="" type="checkbox"/> Adresa	<input type="checkbox"/> Datum_Registrace
<input checked="" type="checkbox"/> Fakulta	
<input type="checkbox"/> Katedra	
<input type="checkbox"/> Zkratka	
<input checked="" type="checkbox"/> Kontaktni_Osoba	
<input checked="" type="checkbox"/> Email	
<input checked="" type="checkbox"/> Telefon	

☐ Return only unique rows

WHERE...

ORDER BY...

Advanced...

SELECT statement:  
SELECT [Nazev], [Adresa], [Fakulta], [Kontaktni\_Osoba], [Email], [Telefon] FROM [Tab\_Skola]

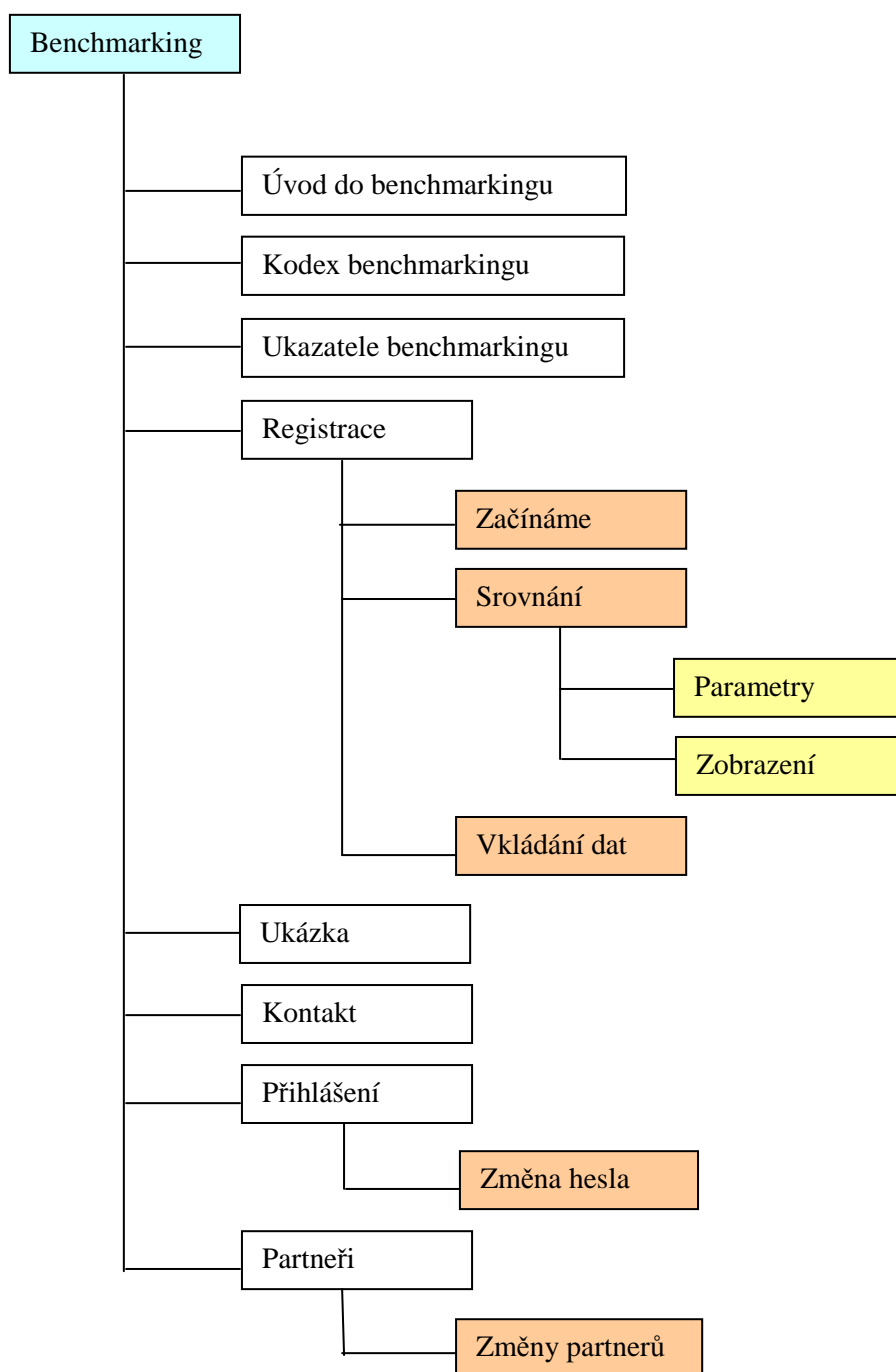
< Previous   Next >   Finish   Cancel

Obr. 30 Výběr tabulky a jejích sloupců

## 7 Mapa stránek, tvorba šablony

### 7.1 Mapa stránek

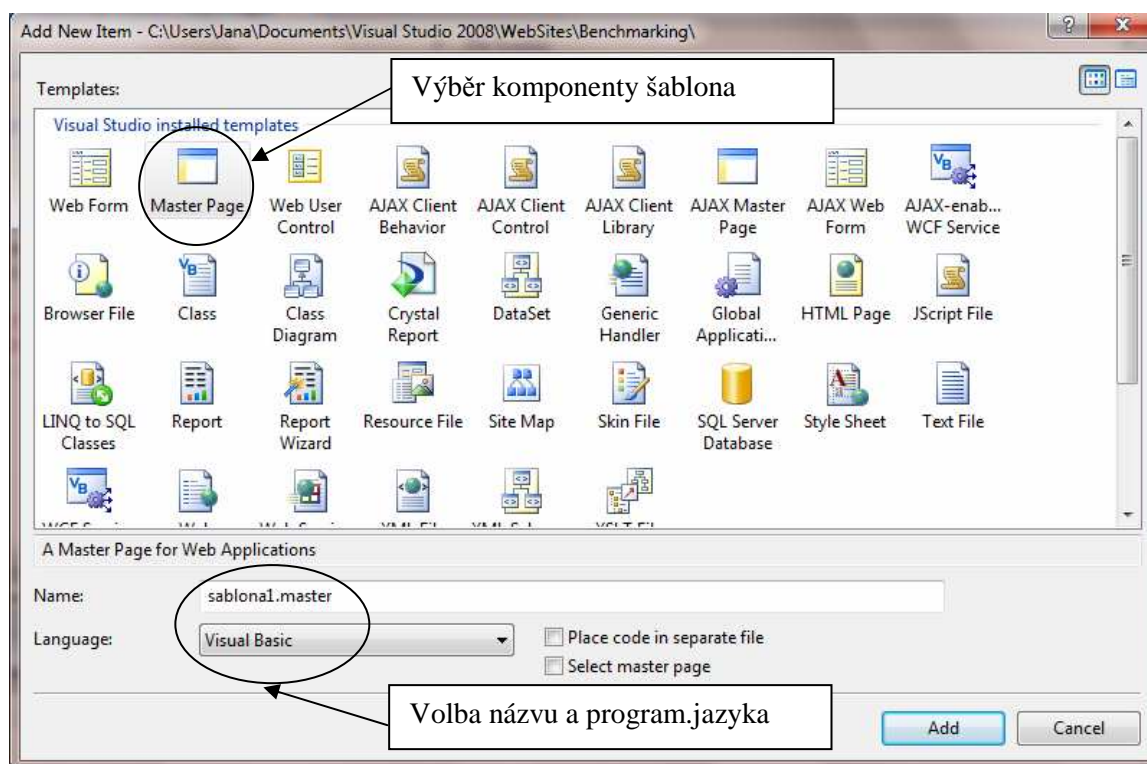
Pro základní orientaci ve struktuře stránek jsem vytvořila jednoduchou mapu stránek, která názorně zobrazuje, do kolika tříd a podtříd budou webové stránky členěny a jak na sebe jednotlivé stránky budou navazovat.



Obr. 31 Mapa stránek

## 7.2 Tvorba šablony

Pro jednotný vzhled jsem vytvořila jednoduchou šablonu, na jejím konceptu budu později vytvářet další stránky naší aplikace. V hlavním menu zvolíme volbu Add New Item, vybereme komponentu Master Page – šablona, zvolíme název a programovací jazyk šablony a dáme přidat, Obr. 32. V tuto chvíli již máme šablonu vytvořenu, nyní upravíme její vzhled.



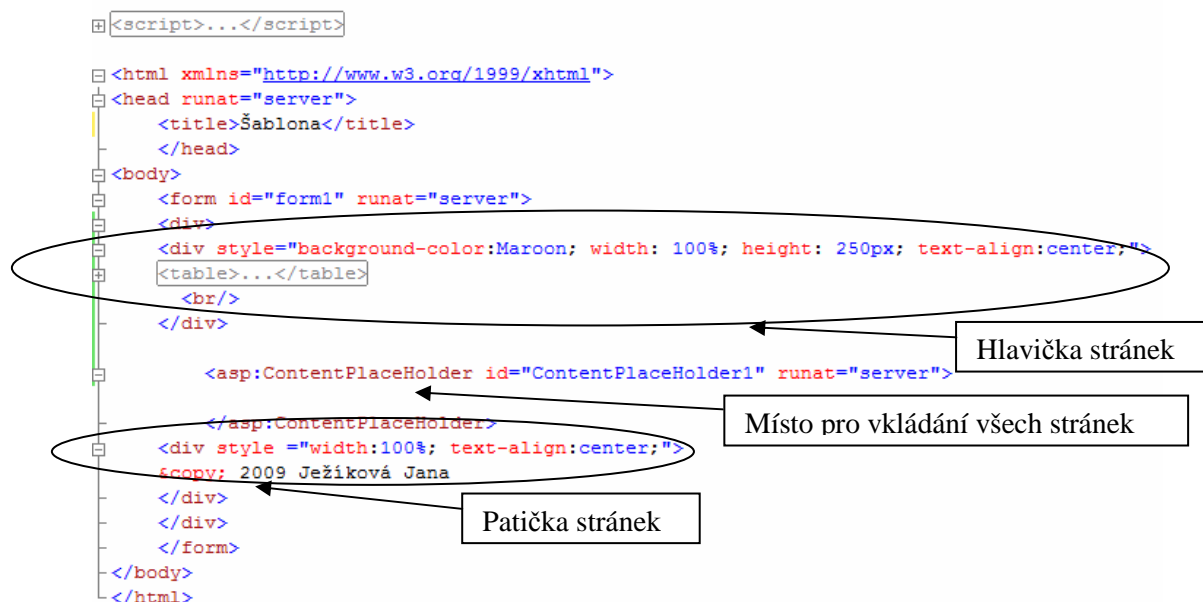
Obr. 32 Přidání šablony

Pro jednotný vzhled stránek jsem na začátek skriptu vložila programový kód, který nám u každé stránky definuje její název, viz. Obr. 33. Ukázka vzhledu je na Obr. 35.

```
<script runat="server">
    Protected Sub Page_PreRender(ByVal sender As Object, ByVal e As System.EventArgs)
        Page.Header.Title = "Benchmarking |" & Page.Header.Title
    End Sub
</script>
```

Obr. 33 Jednotná definice názvů stránek

Šablonu jsme si rozdělili na tři části (hlavička, obsah, patička), viz. Obr. 34, které postupně namodifikujeme. Volbu “Place code in separate file” nevybíráme, jelikož nechceme aby se nám programový kód ukládal do zvláštní složky. Také volby “Select master page” nevyužijeme, jelikož ta slouží k přidání již vytvořené šablony. Tím by naše nově vytvořená šablona převzala vzhled nějaké již dříve vytvořené šablony.



Obr. 34 Části šablony



Uživatel nepřihlášen | Přihlásit

# IS BENCHMARKINGU VYSOKÝCH ŠKOL

[Začneme](#) [Vkládání dat](#) [Srovnání](#) [Uživatelé](#) [Ukazatele](#)

## MENU

[Domovská stránka](#)  
[Benchmarking](#)  
[Kodex benchmarkingu](#)  
[Ukazatele](#)  
[Registrace](#)

## KALENDÁŘ

[duben](#) **květen 2010** [červen](#)

po	út	st	čt	pá	so	ne
26	27	28	29	30	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

## Přihlášení

Uživatelské jméno

Heslo

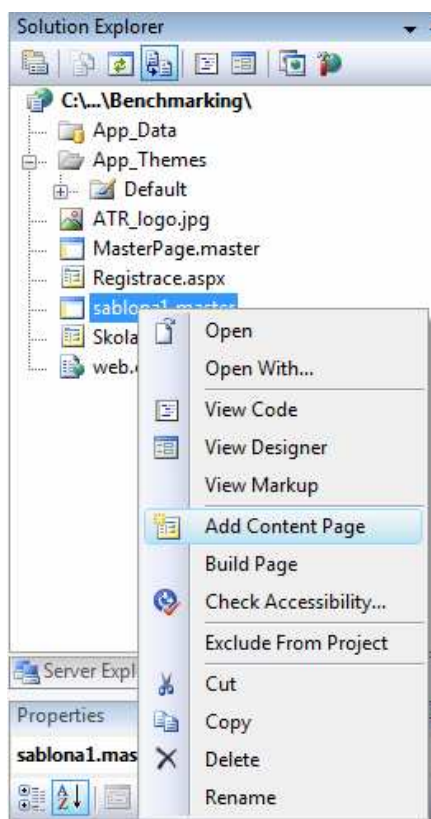
☐ Zapamatovat si mě pro příští přihlášení.

**Přihlásit**

Obr. 35 Vzhled šablony

## 8 Registrace, přihlášení, změna hesla uživatele

Stránka pro registraci je vytvořena na základě šablony. Do této stránky jsem umístila komponenty SqlDataSource, FormView, Button1 a Button2.



Obr. 36 Přidání nové stránky se vzhledem šablony

Pomocí definice Configure Data Source v komponentě SqlDataSource jsem vybrala příslušnou databázi, tabulku, jednotlivé položky a jejich řazení a tím jsem vytvořila spojení s naší databází benchmarkingu, Obr. 37.

```
<asp:SqlDataSource ID="SqlDataSource1" runat="server"
    ConnectionString="<%$ ConnectionStrings:BenchmarkingDatabaseConnectionString %%"
    InsertCommand="INSERT INTO [Tab_Skola] ([Nazev], [Adresa], [Fakulta], [Katedra], [Zkratka],
    [Kontaktni_Osoba], [Email], [Telefon], [Poznamka], [Jmeno_Uzivatele], [Heslo_Uzivatele],
    [Datum_Registrace]) VALUES (@Nazev, @Adresa, @Fakulta, @Katedra, @Zkratka, @Kontaktni_Osoba,
    @Email, @Telefon, @Poznamka, @Jmeno_Uzivatele, @Heslo_Uzivatele, @Datum_Registrace)"
    SelectCommand="SELECT * FROM [Tab_Skola] WHERE ([ID] = @ID)"
    UpdateCommand="UPDATE [Tab_Skola] SET [Nazev] = @Nazev, [Adresa] = @Adresa, [Fakulta] = @Fakulta,
    [Katedra] = @Katedra, [Zkratka] = @Zkratka, [Kontaktni_Osoba] = @Kontaktni_Osoba, [Email] = @Email,
    [Telefon] = @Telefon, [Poznamka] = @Poznamka, [Jmeno_Uzivatele] = @Jmeno_Uzivatele,
    [Heslo_Uzivatele] = @Heslo_Uzivatele, [Datum_Registrace] = @Datum_Registrace WHERE [ID] = @ID">
    <SelectParameters>...</SelectParameters>
    <UpdateParameters>...</UpdateParameters>
    <InsertParameters>...</InsertParameters>
</asp:SqlDataSource>
```

Obr. 37 Připojení dat z databáze

Dále jsem do stránky vložila formulář, jehož vzhled jsem upravila a vložila do něj validátory, které slouží pro zajištění vkládání dat ve správném formátu a pro zajištění vyplnění všech povinných údajů ve formuláři. Tím zajistíme, aby všechny položky z databáze, které mají povinný parametr vložení, byly ve formuláři vyplněny. Také jsem vložila tlačítko pro přidání školy, na kterém jsem definovala proceduru pro odeslání automatického emailu administrátorovi.

## Registrace nového uživatele

### *Podmínky pro registraci*

Tímto se zavazuji, že veškeré níže vyplněné údaje o uživateli jsou správné. V případě lživě uvedených registračních údajů, stejně tak dat týkajících se vedení školy, je správce (tj. Vysoká škola Báňská Technická univerzita Ostrava) oprávněn udělit uživateli peněžní sankci v odpovídající hodnotě za každý fiktivní údaj. Dále se uživatel zavazuje ke kodexu benchmarkingu, jehož plné znění je uvedeno na těchto stránkách, viz. [kodex benchmarkingu](#), výčtově k mlčenlivosti a nešíření osobních a citlivých údajů třetím osobám.

Název	<input type="text"/>
Adresa	<input type="text"/>
Fakulta	<input type="text"/>
Katedra	<input type="text"/>
Zkratka	<input type="text"/>
Kontaktní osoba	<input type="text"/>
Email	<input type="text"/>
Telefon	<input type="text"/>
Poznámka	<input type="text"/>
Jméno uživatele	<input type="text"/>
Datum registrace	<input type="text"/>
<input type="button" value="Přidat uživatele"/>	

Obr. 38 Registrace

Položky email a telefon jsem také zabezpečila regulárními výrazy, které kontrolují správnost zadávaných údajů ve formulářích. Používají se téměř ve všech programovacích jazycích. Existují různé typy regulárních výrazů – nejznámější dvě jsou Perl-compatible regulární výrazy a POSIX



regulární výrazy. V mé práci používám pouze jednoduché regulární výrazy. Na ukázkou zde uvádím tabulku s příklady použití regulárních výrazů.

Tab. 4 Příklady regulárních výrazů

Regulární výraz	Popis
b[au]f	Odpovídá řetězci baf nebo buf.
[0-9].[[:alpha:]]	Odpovídá řetězci, který obsahuje číslici, poté libovolný znak a za ním písmeno.
[a-zA-Z0-9._-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,4}	Jednoduchý výraz pro e-mail
^(\+420)? ?[0-9]{3} ?[0-9]{3} ?[0-9]{3}\$	odpovídá českému telefonnímu číslu v národním či mezinárodním formátu (tj. s předvolbou +420); předpokládá, že bude devíticiferné číslo napsáno bez mezer nebo bude mezera následovat vždy po skupině tří číslic (tj. jediný správný zápis devíticiferného telefonního čísla); devíticifernému číslu může (ale nemusí) předcházet +420

Teprve poté, co administrator přidá uživateli přístupové práva bude moci uživatel do systému přistupovat. Celý programový kód stránky pro registraci je uveden v Příloze 1.

```
Protected Sub SendEmail(ByVal sender As Object, ByVal e As EventArgs)

    Const ToAddress As String = "janickajezecek@seznam.cz"
    Const FromAddress As String = "martinkajezecek@seznam.cz"

    Dim mm As New MailMessage(FromAddress, ToAddress)
    mm.Subject = "Žádost o registraci do benchmarkingu VŠ"
    mm.Body = info
    mm.IsBodyHtml = False

    Dim smtp As New SmtpClient
    smtp.Credentials.GetCredential("smtp.vsb.cz", "25", "network")
    smtp.Send(mm)
    Dim responses As MsgBoxResult
    responses = MsgBox("Žádost o registraci byla úspěšně zaslána. Po schválení  
Response.Redirect("~/Default.aspx")

End Sub
```

Obr. 39 Procedura SendEmail

Formulář pro přihlášení se automaticky zobrazí vždy, pokud chce nepřihlášený uživatel vstoupit do sekce pro registrované. To je zajištěno aplikací asp.net configuration, pomocí správce přístupových práv.

## Přihlášení

Uživatelské jméno

Heslo

☐ Zapamatovat si mě pro příští přihlášení.

**Přihlásit**

Obr. 40 Přihlášení

Dalším formulářem je formulář pro změnu hesla. Je to zajištěno komponentou changepassword, na které jsem modifikovala její design.

## Změna hesla

Změna hesla

Staré heslo

Nové heslo

Zopakovat nové heslo

**Změnit heslo** **Beze změn**

Nové heslo musí mít minimálně 8 znaků.

Obr. 41 Změna hesla

## 9 Seznam škol, úprava a odstranění

Stránka s názvem `Uzivatele.aspx` bude sloužit jako přehled registrovaných škol. Jednotliví uživatelé si zde budou moci vyhledávat další účastníci-se školy. Dále na stejném principu bude pracovat i druhá stránka s názvem `Skola.aspx`, která bude obsahovat úplně ty samé informace, ale také dva další speciální sloupce pro editaci a mazání uživatelů. Tentokrát nebude určena uživateli, nýbrž administrátorovi, který jako jediný má možnost a právo jednotlivé školy editovat, případně odstranit.

### Uživatelé

Hledat školu:

Nazev	Adresa		
ČVUT Praha	Staroměstská 1111	<a href="#">Upravit</a>	<a href="#">Smazat</a>
FS TU Liberec	3.října 444, Liberec	<a href="#">Upravit</a>	<a href="#">Smazat</a>
MTF STU Bratislava	Okrová 458, Bratislava	<a href="#">Upravit</a>	<a href="#">Smazat</a>
SjF STU Bratislava	Bratislava	<a href="#">Upravit</a>	<a href="#">Smazat</a>
SjF ŽU Žilina	6.ledna, Žilina	<a href="#">Upravit</a>	<a href="#">Smazat</a>
Technická univerzita Plzeň	Plzeňská 356, Plzeň	<a href="#">Upravit</a>	<a href="#">Smazat</a>
TnUAD Trnava	835 Trnava	<a href="#">Upravit</a>	<a href="#">Smazat</a>
Univerzita Tomáše Bati	Zlín 485	<a href="#">Upravit</a>	<a href="#">Smazat</a>
Vysoká škola Báňská Technická univerzita Ostrava	17.listopadu, Ostrava	<a href="#">Upravit</a>	<a href="#">Smazat</a>
Vysoká škola ekonomická Pelhřimov	25.května 140, Pelhřimov 1, 739 11	<a href="#">Upravit</a>	<a href="#">Smazat</a>
Vysoká telekomunikační škola	Přerovská 123, Přerov 75864	<a href="#">Upravit</a>	<a href="#">Smazat</a>
Vysoké učení technické Brno	1.máje, Brno-střed	<a href="#">Upravit</a>	<a href="#">Smazat</a>

Obr. 42 Náhled stránky `Skola.aspx`

Abych zbytečně dvakrát nepopisovala tu samou stránku, proto zde uvedu pouze popis programového kódu pro stránku určenou administrátorovi, `Skola.aspx`. V příloze 2 a příloze 3 jsou uvedeny kompletní programové kódy obou stránek. V komponentě `GridView` jsou definovány sloupce, které mají být zobrazeny a pomocí další komponenty `SqlDataSource` je realizováno propojení těchto sloupců s sql databází. Pro realizaci vyhledávání je zde jednoduchý `TextBox`, který je propojen opět s `SqlDataSource-název`. Nakonec je zde tlačítko, po jehož stisknutí se vyhledají pouze školy se zadaným názvem. Ukázka využití vyhledávání je umístěna na Obr. 43.

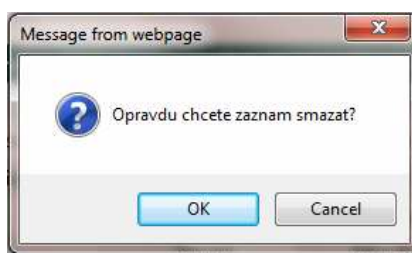
### Uživatelé

Hledat školu: vysok

Nazev	Adresa		
Vysoká škola Báňská Technická univerzita Ostrava	17.listopadu, Ostrava	<a href="#">Upravit</a>	<a href="#">Smazat</a>
Vysoká škola ekonomická Pelhřimov	25.května 140, Pelhřimov 1, 739 11	<a href="#">Upravit</a>	<a href="#">Smazat</a>
Vysoká telekomunikační škola	Přerovská 123, Přerov 75864	<a href="#">Upravit</a>	<a href="#">Smazat</a>
Vysoké učení technické Brno	1.máje, Brno-střed	<a href="#">Upravit</a>	<a href="#">Smazat</a>

Obr. 43 Ukázka vyhledávání

V případě že uživatel nezadá pro vyhledávání žádný výraz, vrátí se mu všechny školy. Sloupeček pro editaci je realizován pomocí HyperLinkField, kde je jako text uvedeno Upravit, DataNavigateUrlFields představuje omezení (pro editaci uživatele s identifikačním číslem 1 se nám zobrazí stránka uvedená v DataNavigateUrlFormatString s údaji o uživateli s identifikačním číslem 1, kdybychom zde toto omezení neuvedli, server by nevěděl, kterého uživatele chceme editovat, o kterém uživateli nám má zobrazit editační data) a DataNavigateUrlFormatString nám udává odkaz na stránku Editace.aspx. Poslední sloupeček pro mazání je realizován tlačítkem, na kterém je definována procedura po stisknutí, která nám zobrazí dialogové okno s dotazem zda opravdu chceme uživatele smazat. Tímto je zamezeno překliknutí, neúmyslnému smazání uživatele.



Obr. 44 Potvrzení smazání uživatele

Pro editaci a smazání školy ze seznamu a z databáze byla vytvořena webová stránka Editace.aspx, která je ve svém principu velmi podobná stránce pro registraci. Obsahuje tentýž formulář, tutéž definici sloupců, která však není umístěna v sekci InsertItemTemplate, jak je tomu u registrace, nýbrž ale v sekci EditItemTemplate. Opět jsou zde validátory, jelikož i při editaci musí být požadované sloupce zadány ve správném formátu a musí být zajištěno vyplnění všech nutných sloupců. Nakonec jsou zde opět dvě tlačítka, která uloží případně neuloží změny do databáze a přesměrují se zpět na stránku Skola.aspx. Programový kód je zobrazen v příloze 4.

## Editace uživatele

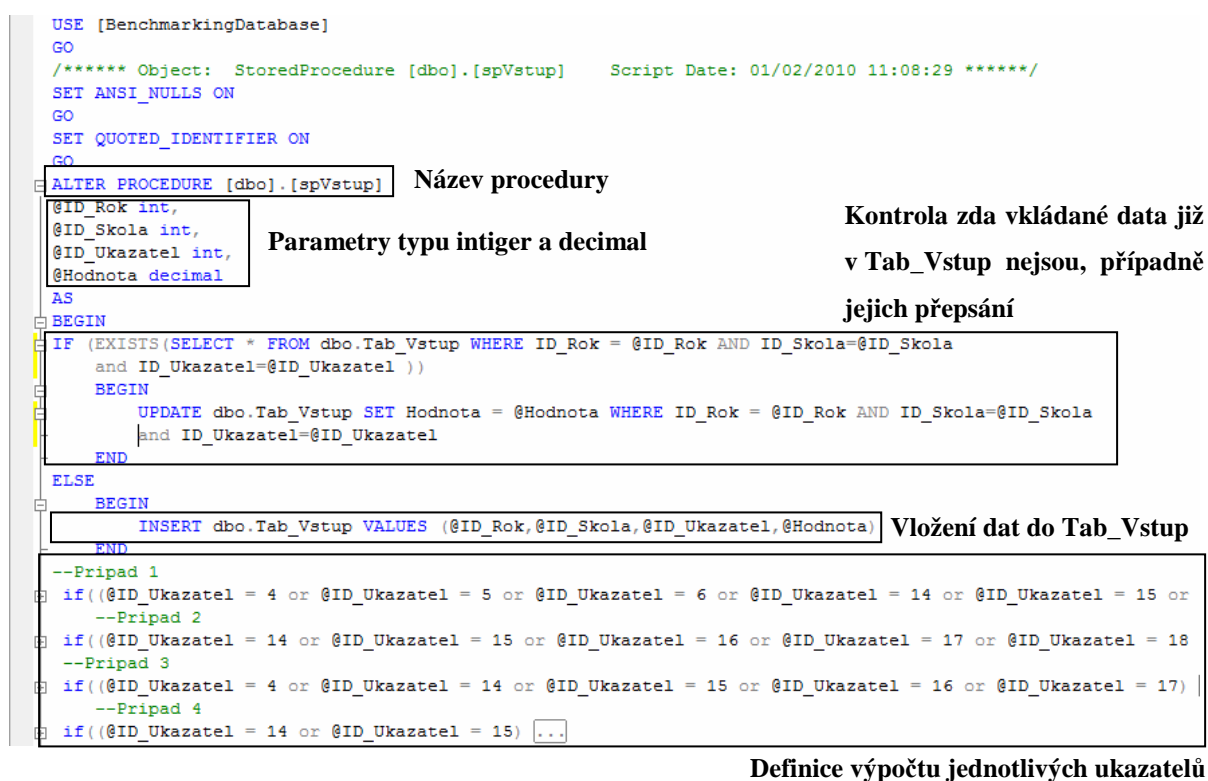
Název	Vysoká škola Báňská Technická univerzita Ostrava
Adresa	17.listopadu, Ostrava
Fakulta	Strojní
Katedra	352
Zkratka	352-FS-VŠB Ostrava
Kontaktní osoba	Radim Farana
Email	radim.farana@vsb.cz
Telefon	777111222
Poznámka	
Jméno uživatele	RF
Datum registrace	1.1.2004 0:00:00
<input type="button" value="Uložit změny"/> <input type="button" value="Zrušit"/>	

Obr. 45 Náhled editace uživatele

## 10 Vkládání dat

V minulých kapitolách jsme vkládání dat do databáze již využili ve stránkách uživatel, škola a registrace.aspx. Ve všech případech jsem použila jednoduché designové připojení k databázi. Nyní je potřeba vytvořit stránku pro vkládání výpočetních údajů ukazatelů. Jelikož při vkládání dat do databáze, do Tab\_Vstup musí docházet také k jejich přepočtu a tím k naplnění databázové tabulky Tab\_Vystup je potřeba nějakým způsobem tento přepočet ošetřit.

Na straně databáze jsem vytvořila novou proceduru sp\_Vstup, která reaguje na vkládání dat do tabulky Tab\_Vstup.



Obr. 46 Procedura sp\_Vstup

Pro každý ukazatel jsem vytvořila definici výpočtu, jelikož jsem nenašla žádné společné znaky ve výpočtech. S tímto řešením však vyvstává nevýhoda při přidání nového ukazatele, jelikož administrátor bude muset v proceduře `sp_Vstup` definovat výpočet nového ukazatele.

Zde je zobrazen programový kód výpočtu ukazatele, jehož ID je 4. Nejprve se ověří zda v `Tab_Vstup` existují pro požadovaný rok a školu, konkrétní data nezbytná pro jeho výpočet. Jestliže neexistují pak se tento případ uzavře a testuje se další z případů.

Pokud však požadovaná data existují, načtou se do deklarovaných pomocných proměnných, provede se definovaný výpočet, který se uloží do `Tab_Vystup`. Pokud v `Tab_Vystup` již údaj pro

příslušný rok a školu existuje, přepíše se příslušný řádek, pokud však zmiňovaný údaj v tabulce není, vytvoří se nový řádek, do kterého se vypočtená hodnota uloží.

```
--Případ 4
if (@ID_Ukazatel = 14 or @ID_Ukazatel = 15)
AND EXISTS(SELECT * FROM dbo.Tab_Vstup WHERE ID_Rok = @ID_Rok AND ID_Skola=@ID_Skola and ID_Ukazatel=14 )
AND EXISTS(SELECT * FROM dbo.Tab_Vstup WHERE ID_Rok = @ID_Rok AND ID_Skola=@ID_Skola and ID_Ukazatel=15 )
)
BEGIN
DECLARE @A6 decimal, @A7 decimal, @V4 decimal
SELECT @A6=Hodnota FROM dbo.Tab_Vstup WHERE ID_Rok = @ID_Rok AND ID_Skola=@ID_Skola and ID_Ukazatel=14
SELECT @A7=Hodnota FROM dbo.Tab_Vstup WHERE ID_Rok = @ID_Rok AND ID_Skola=@ID_Skola and ID_Ukazatel=15
SET @V4 = @A6+(0.5*@A7)
IF (EXISTS(SELECT * FROM dbo.Tab_Vystup WHERE ID_Rok = @ID_Rok AND ID_Skola=@ID_Skola
and ID_Vypocet_ID_Ukazatel=4 ))
BEGIN
UPDATE dbo.Tab_Vystup SET Hodnota = @V4 WHERE ID_Rok = @ID_Rok AND ID_Skola=@ID_Skola
and ID_Vypocet_ID_Ukazatel=4
END
ELSE
BEGIN
INSERT dbo.Tab_Vystup(ID_Rok, ID_Skola, ID_Vypocet_ID_Ukazatel, Hodnota)
VALUES (@ID_Rok, @ID_Skola, 4, @V4)
END
END
```

Obr. 47 Definice výpočtu ukazatele s identifikačním číslem 4

Poté jsem vytvořila novou stránku Vkladani.aspx. Zde jsem umístila komponenty DropDownList a SqlDataSource, které jsem spolu provázala. Dále jsem do stránky vložila Label s názvem Hodnota a TextBox, který bude sloužit k vložení číselné hodnoty. Nakonec jsem vložila tlačítko Vložit, na kterém je definovaná metoda po kliknutí. Při vyvolání této metody dojde k spuštění procedury sp\_Vstup a k aktualizaci stránky pro nové vkládání dat. Kompletní programový kód stránky je umístěn v Příloze 5.

## Vkládání dat

Rok

Instituce

Počet úspěšných studentů 1.ročníku Bc. studia k 31.10.  
 Počet úspěšných studentů 2.ročníku Bc. studia k 31.10.  
 Počet úspěšných studentů 1.ročníku Mgr. studia k 31.10.  
 Počet studentů doktorského studia k 31.10.  
 Počet studentů přijatých do 1.ročníku Bc. studia  
 Počet studentů přijatých do 1.ročníku Mgr. studia  
 Počet úspěšných studentů 3.ročníku Bc. studia k 31.10  
 Počet úspěšných studentů u SZZ u Bc. studia v 1.termínu  
 Počet úspěšných studentů u SZZ u Bc. studia v 2.termínu  
 Počet úspěšných studentů u SZZ u Mgr. studia v 1.termínu  
 Počet úspěšných studentů u SZZ u Mgr. studia ve 2.termínu  
 Počet studentů přijatých na doktorské studium  
 Počet docentů na HPP  
 Počet docentů na VPP  
 Počet profesorů na HPP  
 Počet profesorů na VPP  
 Počet V-V pracovníků na HPP  
 Počet V-V pracovníků na VPP  
 Počet odborných asistentů na HPP  
 Počet odborných asistentů na VPP  
 Počet asistentů a lektorů na HPP  
 Počet asistentů a lektorů na VPP  
 Počet ostatních pracovníků  
 Věkový součet všech profesorů

Ukazatel

Hodnota

Obr. 48 Stránka Vkladani.aspx

## 11 Nástroje pro tvorbu grafů

### 11.1 Fusion charts

- jednoduchý program, který je zcela zdarma
- lze použít pro různé programy a programovací jazyky (ASP.NET, ASP, PHP, JSP, HTML)
- různé druhy grafů, včetně množství 3D grafů

### 11.2 Google Chart API

- extrémně jednoduché na používání
- graf na stránce je obrázek
- <http://code.google.com/intl/cs-CZ/apis/chart/>
- není omezené pro jeden skriptovací jazyk
- opensource
- editace parametrů grafu přímo v adrese
- parametry:
  - typ grafu
  - data grafu
  - velikost grafu
  - barva grafu
  - popisky os
- není potřeba znát žádné webové technologie, např. XHTML, XML
- jednotlivé parametry jsou odděleny znakem &
- ukázka – rozbor kodu:  
<http://chart.apis.google.com/chart?cht=p3&chd=s:Hello&chs=250x100&chl=May|Jun|Jul|Aug|Sep|Oct> (chs=250x100 – rozměry vygenerovaného grafu v pixelech, chd=s:Ahojjj – data, která

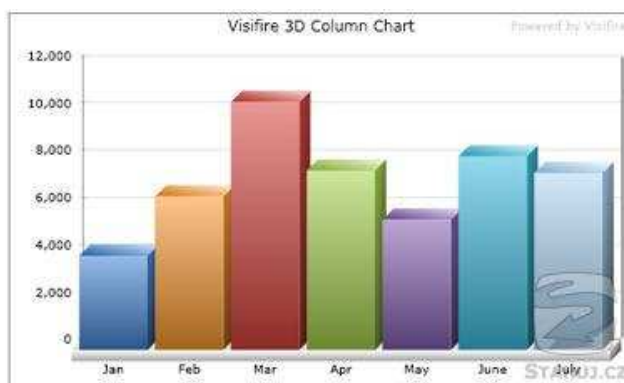
graf zobrazuje, cht=p3 – typ grafu, p3 je kruhový graf, chl=May|Jun|Jul|Aug|Sep|Oct – štítky popisují, co vynesená data představují )

## 11.3 Xceed Chart for ASP.net

- komponent pro .NET napsaný v jazyce C#
- obsahuje panely nástrojů a gridy, návrh grafů
- více jak 20 editorů pro nastavení vlastností grafů
- 14 základních typů grafů
- zobrazení více grafů najednou
- přibližná cena licence 8568 Kč (listopad 2009)

## 11.4 Visifire

- pro správu a design grafů, opensource
- funguje díky Microsoft Silverlight – musí být nainstalován
- jednoduché rozhraní, nezávislost na serverové technologii.
- možno použít s ASP, ASP.Net, PHP, JSP, ColdFusion, Ruby on Rails nebo HTML
- animovaná podoba grafů ve 2D i 3D
- možnost snadného vložení na webové stránky



Obr. 49 Graf v programu Visifire [[http://www.stahuj.centrum.cz/direct/iR/visifire/\\_detail.jpg](http://www.stahuj.centrum.cz/direct/iR/visifire/_detail.jpg)]

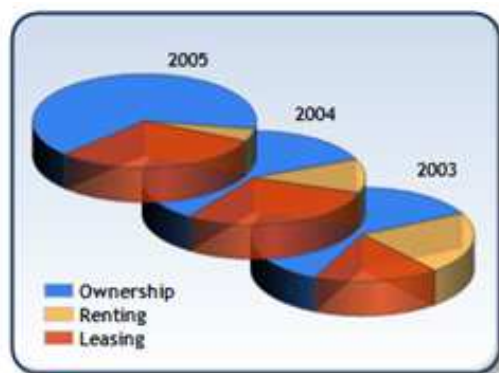


## 11.5 Tee chart

- komponenta napsaná v jazyce C pro prostředí .NET
- zobrazení 2D I 3D grafů
- obsahuje průvodce i manual
- umožňuje export do HTML, JPG, PNG, WMF, EMF, PCX, VML, BMP, GIF, CVS
- placená licence 13363 Kč (listopad 2009)

## 11.6 Microsoft Chart Controls

- ovládací prvek od Microsoftu
- opensource
- snadná tvorba grafů
- široká škála typů grafů (koláčové, finanční, bodové, sloupcové a další)
- 3D zobrazení
- podpora cachování a ukládání obrázků na disk

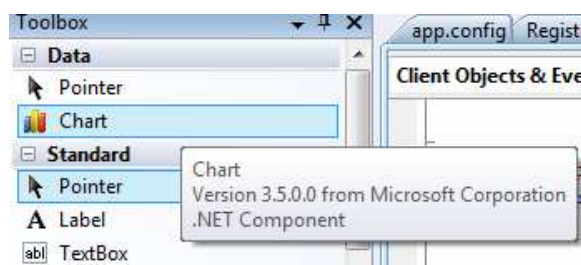


Obr. 50 Ukázka grafu v MSChart [<http://weblogs.asp.net/scottgu/archive/2008/11/24/new-asp-net-chartingcontrol-It-asp-chart-runat-quot-server-quot-gt.aspx>]

Z výše uvedených možností jsem si k mé práci vybrala poslední produkt, tj. Microsoft Chart Controls, jelikož se mi zdá velice propracovaný, s vysokým výběrem grafů a zobrazení. Jako největší nevýhodu zde vidím anglický manuál, který může být pro spoustu uživatelů velkou překážkou.

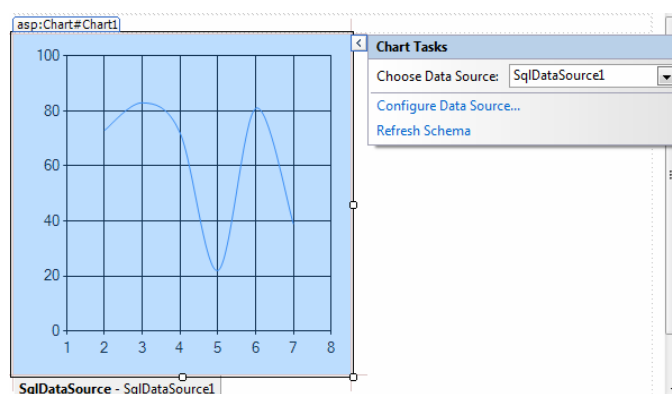
### 11.6.1 Vykreslování grafů v MS Chart Control

Pro tvorbu grafů jsem si vybrala komponentu Microsoft Chart Controls, která je opensource a která je přímo dostupná na webových stránkách. Po dokončení instalace postačí restartovat Microsoft Visual Studio a již se nám v nabídce nástrojů zobrazí komponenta Chart pro vykreslování grafů.



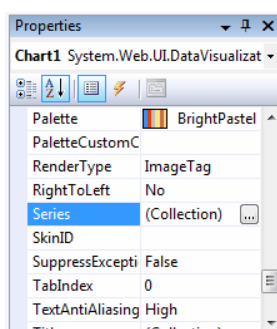
Obr. 51 Komponenta Chart v nabídce nástrojů

Vytvořila jsem si novou stránku Graf.aspx, do které jsem umístila komponentu Chart. Stejně jako u komponent FormView, GridView, DropDownList a dalších tak i zde se data pro graf připojují velice jednoduše, pomocí připojovacího retězce, viz. Obr. 52.



Obr. 52 Připojení databáze k grafu

Nyní již máme vytvořený graf, propojený s databází. Aby se nám ale zobrazila potřebná data, musíme nastavit, které z dat budou umístěny na ose x, které na ose y, jaký typ grafu chceme zobrazit, atd.. K tomu slouží Series, kterou najdeme ve vlastnostech grafu.



Obr. 53 Menu vlastností komponenty chart

## 11.6.2 Tabulkové a grafické srovnání

Pro srovnání organizací jsem vytvořila 2 stránky. V jedné, s názvem ZobrazeniDat.aspx, si uživatel vybere kritéria pro srovnání, poté co stiskne tlačítko vybrat data se otevře druhá stránka Graficke.aspx, ve které se mu zobrazí tabulka žádaných dat a výsledný graf.

### Tabulkové srovnání

Obr. 54 Kritéria pro srovnání

Tabulka je řešena komponentou griedview, která načítá data z sqldatasource z databázového pohledu view\_graf.

### Grafické srovnání

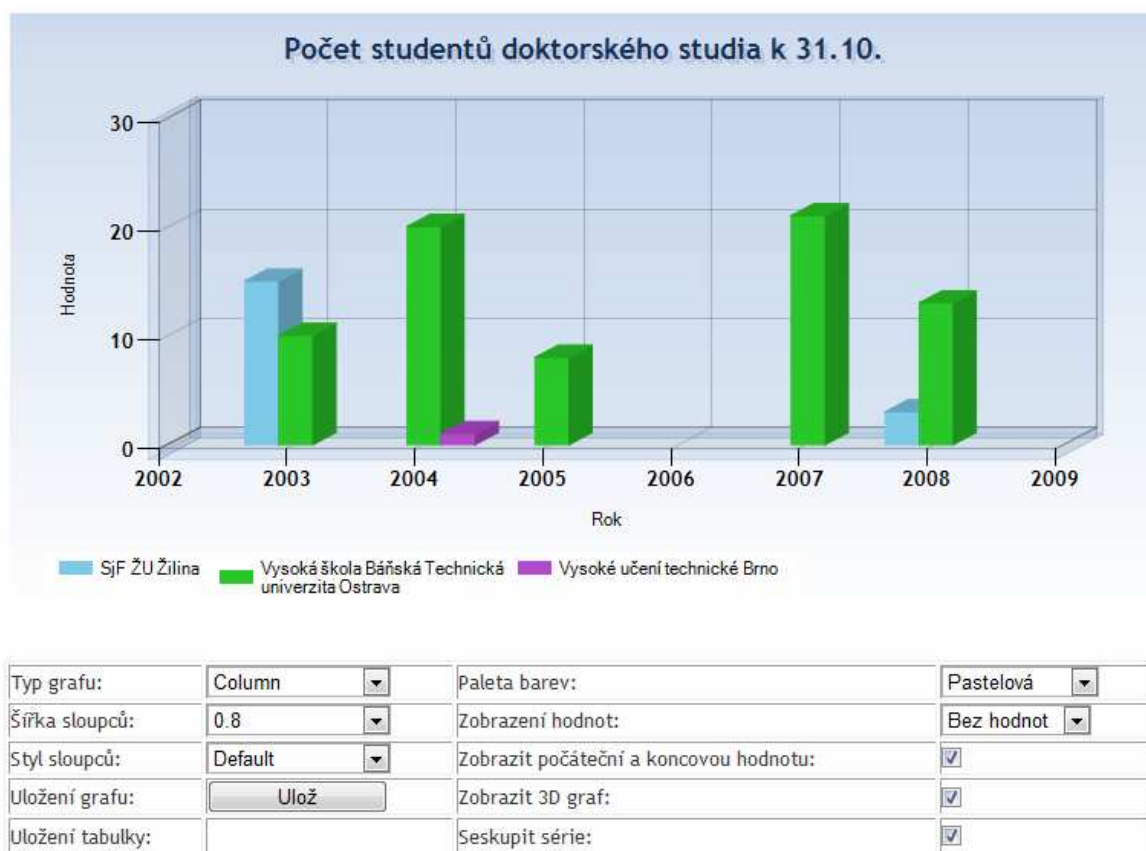
Data pro graf:

Rok	Nazev	Hodnota
2003	Vysoká škola Báňská Technická univerzita Ostrava	10
2003	SjF ŽU Žilina	15
2004	Vysoká škola Báňská Technická univerzita Ostrava	20
2004	Vysoké učení technické Brno	1
2005	Vysoká škola Báňská Technická univerzita Ostrava	8
2007	Vysoká škola Báňská Technická univerzita Ostrava	21
2008	SjF ŽU Žilina	3
2008	Vysoká škola Báňská Technická univerzita Ostrava	13

Obr. 55 Tabulka vybraných dat

Aby data pro graf nebyla statická, bylo nutno toto ošetřit programovým kódem. Vytvořila jsem pohledy graf, pocet\_zobr\_skol a ukazatel. Tyto pohledy jsou dále odesílány pomocí sqldatasource do příslušných sqldatareadrů. Pomocí těchto sqldatareadrů je pak realizováno načítání názvu grafu, zobrazení legendy a popisků os. Pro úpravu jsem zde přidala také tabulku, která nabízí

různé modifikace úpravy grafu jako je zobrazení 3D grafu, změna typu grafu, stylu sloupců, šířky sloupců, zobrazení hodnot sloupců a uložení grafu. Celý programový kód stránky je uveden v příloze 9.



Obr. 56 Výsledné zobrazení grafu

## 12 Závěr

Cílem diplomové práce bylo vytvořit webový informační systém pro podporu benchmarkingu vysokých škol. Nejprve jsem řešila terminologii, seznamovala se s pojmy týkajícími se tématu benchmarking. Dále jsem benchmarking zkoumala z pohledu vysokých škol. Z dodaných podkladů minulých sběrů dat jsem vytvořila seznam sbíraných dat a používaných ukazatelů, jejichž definice výpočtů jsem zde zdokumentovala.

Následně jsem se zabývala databázovými systémy, jejich výhodami, nevýhodami a volbou databázového systému pro tvorbu benchmarkingu. Předě mnou problém benchmarkingu řešil i Martin Šlosarčík [Šlosarčík M.,2009]. Na základě přání vedoucího práce jsem tuto databázovou strukturu převzala s tím, že jsem jí mírně upravila a přepracovala do databáze SQL Serveru 2008, který jsem si pro databázovou tvorbu vybrala. Poté jsem databázi naplnila zkušebními daty a ověřila její funkčnost pomocí několika dotazů.

Tímto skončila základní realizace databáze a já začala navrhovat strukturu aplikační vrstvy a realizovala její pilotní část. Pro tvorbu jsem si vybrala ASP.NET, se kterým jsem již byla seznámena v jiném předmětu a jehož prostředí mi bylo známé a pro tvorbu této aplikační vrstvy vhodné. Začala jsem tvořit jednotlivé stránky webové aplikace. Nejprve jsem vytvořila stránku pro registraci, kterou jsem ošetřila validátory a regulárními výrazy pro email a telefon a po jejímž vyplnění bude administrátorovi odeslán automatický email o nové registraci uživatele. Poté co administrator dostane email s žádostí o registraci, vytvoří nový uživatelský účet a zašle uživateli zpět email o aktivaci jeho účtu. Dále jsem vytvořila stránku uživatelů, kde lze pohodlně vyhledávat, upravovat, případně mazat uživatele z databáze. Poté jsem vytvořila stránku pro vkládání dat, kterou jsem musela ošetřit procedurou pro přepočítání dat na jednotlivé ukazatele. Také jsem vytvořila stránky pro přihlášení, změnu hesla uživatele a několik stránek týkajících se úvodu do světa benchmarkingu.

Nakonec jsem vyhledala možnosti pro tvorbu grafů, vybrala si ovládací prvek Microsoft Chart Controls, který jsem nainstalovala a naučila se s ním základní úkony. Posledními stránkami, které jsem vytvářela byly stránky pro srovnání organizací. Poté, co si uživatel vybere kritéria pro srovnání a stiskne příslušné tlačítko mu bude zobrazena stránka s výslednou tabulkou a grafem. Tento graf je možno pomocí tlačítka uložit jako obrázek ve formátu jpeg.

Pro jednotný vzhled informačního systému jsem vytvořila šablonu, do které jsem vložila údaje o přihlášení uživatele, vodorovné menu pro registrované uživatele a svislé menu pro všechny návštěvníky stránek. Na závěr jsem pomocí kaskádových stylů doupřipravila design informačního systému do konečné podoby a ověřila funkčnost stránek v několika internetových prohlížečích.

Myslím si, že struktura databáze a její realizace v sql serveru, stejně tak jako zvolení technologie asp.net pro tvorbu aplikační vrstvy byly správné. Pro další zdokonalení informačního systému bych navrhovala úpravu vstupní a výstupní procedury, aby při vložení nového ukazatele nemusel administrator přidávat do této procedury postup výpočtu tohoto ukazatele. Dalším vylepšením by mohl být export výstupní tabulky do excelovského souboru a vylepšení designu stránek.

## Použitá literatura

Businessinfo. *Benchmarking*. [on-line]. cit. 2009-03-15. Dostupný z webu <URL: <http://www.businessinfo.cz/cz/clanek/male-a-stredni-odnikani/benchmarking/1000503/38490/>>

Česká společnost pro jakost. *Benchmarkingový server*. [on-line]. cit. 2008-03-05. Dostupný z webu <URL: <http://www.benchmarking.cz/>>

Czechinvest. *Agentura pro podporu podnikání a investic*. [on-line]. cit. 2009-03-20. Dostupný z webu <URL: <http://www.czechinvest.org/benchmarking>>

Databázový svět. *Vyvíjíme databázový a informační systém*. [on-line]. cit. 2009-04-04. Dostupný z webu <URL: <http://dbsvet.cz/view.php?cisloclanku=2004050501>>

Duthie Andrew G. *Microsoft ASP.NET krok za krokem*. Brno: Mobil Media a.s. 2003 Brno, 511s. ISBN 80-86593-33-9.

Fantasmag net. *Pracujeme s databází v ASP.NET*. [on-line]. cit. 2009-04-05. Dostupný z webu <URL: <http://www.fantomasmag.net/clanky/pracujeme-s-databazi-v-asp-net.aspx>>

Farana, R. *Databáze – speciální postupy*. 1. vyd. Praha : ČSVTS, SmSVTSaP, KAKI, 2006, 170 s. ISBN 80-02-01876-1.

Farana, R. *Podklady pro benchmarking strojních fakult vysokých škol*. [MS-Excel]. Datum poslední změny 2008-02-04 .

Google chart API. *Více informací o Google chart*. [on-line]. cit. 2009. Dostupný z webu <URL: <http://excelplus.net/news.php?readmore=44>>

Hotek M. *Microsoft SQL Server 2008 krok za krokem*. Brno: Computer Press, a.s Brno, 2009. 488s. ISBN 978-80-251-2466-6.

Interval.cz *Grafy v ASP.NET – rady a návody*. [on-line]. cit. 2009-04-05. Dostupný také z webu <URL: <http://interval.cz/clanky/grafy-v-asp-net/>>

Krátký M., *Tvorba informačních systémů – podpora přednášek* [on-line]. cit. 2009. Dostupný z webu <URL: <http://www.cs.vsb.cz/kratky/courses/2007-08/tis/>>

Lacko L'uboslav. *Úvod do novinek SQL Serveru 2005*. 110s. [PDF]

Microsoft AP.NET. *SQL Server 2005 Express for Beginners* [on-line]. cit. 2009. Dostupný z webu  
<URL: <http://www.asp.net/learn/sql-videos/>>

Microsoft ASP.NET. *Videos for ASP.NET 2.0 Beginners* [on-line]. cit. 2009. Dostupný z webu  
<URL: <http://www.asp.net/learn/videos/>>

O webu. *ASP.NET 2.0 – návody, postupy, tipy a triky*. [on-line]. cit. 2009-04-05. Dostupný z webu  
<URL: <http://www.owebu.cz/skripty/vypis.php?clanek=1475>>

Payne Ch. *Naučte se ASP.NET za 21 dní*. Praha: Computer Press, a.s Praha, 2002. 765s. ISBN 80-7226-605-5.

Plecháč V. *Databáze na webu*. Praha: GComp, Praha 2000. 248s. ISBN 80-85649-82-9

Šíma F., Vilímek D. *Microsoft Visual Studio.NET praktické programování krok za krokem*. Praha: Grafa Publishing, a.s Praha 7, 2006, 256s. ISBN 80-247-1418-3.

Šlosarčík, M. *Informační podpora benchmarkingu vysokých škol. Semestrální projekt 2*. Ostrava: VŠB-TU Ostrava [PDF], datum poslední změny: 2009-02-16. Vedoucí: Farana,R..

Vbnet. *Úvod do ASP.NET 2.0* [on-line]. cit. 2009-04-05. Dostupný z webu <URL: [http://www.vbnet.cz/clanek--44-zaciname\\_s\\_asp\\_net\\_dil\\_1\\_uvod\\_do\\_asp\\_net\\_2\\_0.aspx](http://www.vbnet.cz/clanek--44-zaciname_s_asp_net_dil_1_uvod_do_asp_net_2_0.aspx)>

Visifire. *Grafy zdarma, silverlight*. [on-line]. cit. 2009. Dostupný z webu <URL: <http://www.nabito.net/skvele-silverlight-grafy-zdarma>>

Vzdělávací centrum pro veřejnou správu. *Benchmarkingova iniciativa*. [on-line]. cit. 2009-03-15. Dostupný z webu <URL: <http://www.benchmarking.vcvscr.cz/>>

Web tip. *ASP pro začátečníky*. [on-line]. cit 2009. Dostupný z webu <URL: [http://www.webtip.cz/art/wt\\_tech\\_asp/serial\\_asp\\_1.html](http://www.webtip.cz/art/wt_tech_asp/serial_asp_1.html)>

Whalen E., Garcia M., Patel B., Misner S., Isakov V. *Microsoft SQL Server 2005 Velký průvodce administrátora*. Brno: Computer Press,a.s Brno, 2008. 1080s. ISBN 978-80-251-1949-5.

Xceed Chart for ASP.net *Tvorba grafů*. [on-line]. cit. 2009. Dostupný z webu <URL: <http://www.svetsoftware.cz/xceed-chart-for-asp-net>>

Žídek, Adam. *Databázová podpora měření*. VŠB-TUO, katedra ATŘ-352 VŠB-TUO, 2005. 36 s. Bakalářská práce, vedoucí: Farana, R.



## Seznam příloh

Příloha 1: Programový kód stránky Registrace.aspx

Příloha 2: Programový kód stránky Uživatele.aspx

Příloha 3: Programový kód stránky Skola.aspx

Příloha 4: Programový kód stránky Editace.aspx

Příloha 5: Programový kód stránky Vkladani.aspx

Příloha 6: Programový kód stránky Login.aspx

Příloha 7: Programový kód stránky Benchmarking.aspx

Příloha 8: Programový kód stránky ZobrazeniDat.aspx

Příloha 9: Programový kód stránky Graficke.aspx

Příloha 10: Programový kód stránky default.aspx

Příloha 11: Programový kód stránky masterpage.master

Příloha 12: Manuál pro práci s IS Benchmarkingu

Příloha 13: Disk CD-Rom s aplikací “IS Benchmarkingu”